# Massachusetts Institute of Technology

Woods Hole Oceanographic Institution

Joint Program
in Oceanography/
Applied Ocean Science
and Engineering

1930

DOCTORAL DISSERTATION

*Terrain-Relative Navigation for Autonomous
Underwater Vehicles*

by

Diane E. Di Massa

DTIC QUALITY INSPECTED 4

June 1997

19980505 059

MIT/WHOI

97-22

Terrain-Relative Navigation for Autonomous Underwater Vehicles

by

Diane E. Di Massa

Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

and

Woods Hole Oceanographic Institution
Woods Hole, Massachusetts 02543

June 1997

DOCTORAL DISSERTATION

Approved for publication; distribution unlimited.

Approved for Distribution:

Timothy K. Stanton, Chair
Department of Applied Ocean Physics and Engineering
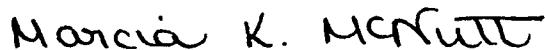
Marcia K. McNutt
MIT Director of Joint Program
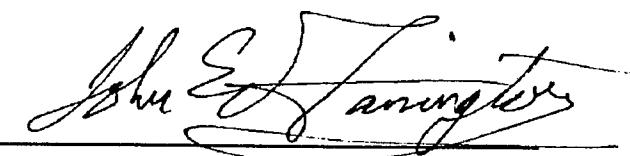
John W. Farrington
WHOI Dean of Graduate Studies

Terrain-Relative Navigation for Autonomous Underwater Vehicles

by

Diane E. Di Massa

B.S., Massachusetts Institute of Technology (1989)
M.S., Massachusetts Institute of Technology (1989)
Mech.Eng., Massachusetts Institute of Technology (1992)

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and the

WOODS HOLE OCEANOGRAPHIC INSTITUTION

May, 1997

© 1997 Diane E. Di Massa. All rights reserved.
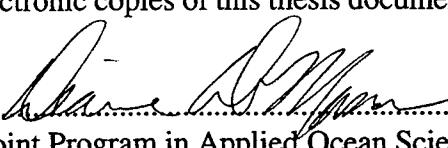
Author ....
Joint Program in Applied Ocean Science and Engineering
Massachusetts Institute of Technology/Woods Hole Oceanographic Institution
May 28, 1997

Certified by ....
Dr. W. Kenneth Stewart, Jr.
Thesis Supervisor

Accepted by ....
Professor Henrik Schmidt
Acting Chair, Joint Committee for Applied Ocean Science and Engineering
Massachusetts Institute of Technology/Woods Hole Oceanographic Institution

Terrain-Relative Navigation for Autonomous Underwater Vehicles

by

Diane E. Di Massa

Submitted to the Department of Ocean Engineering on May 28,
1997, in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Applied Ocean Science and Engineering

Abstract

Navigation is a key technology for autonomous underwater vehicles (AUVs), and currently, it limits potential and existing vehicle capabilities and applications. This thesis presents a terrain-relative navigation system for AUVs that does not require the deployment of acoustic beacons or other navigational aids, but instead depends on a supplied digital bathymetric map and the ability of the vehicle to image the seafloor.

At each time step, a bathymetric profile is measured and compared to a local region of the supplied map using a mean absolute difference criterion. The region size is determined by the current navigation uncertainty. For large regions, a coarse-to-fine algorithm with a modified beam search is used to intelligently search for good matches while reducing the computational requirements. A validation gate is defined around the position estimate using the navigation uncertainty, which is explicitly represented through a covariance matrix. A probabilistic data association filter with amplitude information (PDAFAI), grounded in the Kalman Filter framework, probabilistically weights each good match that lies within the validation gate. Weights are a function of both the match quality and the size of the innovation. Navigation updates are then a function of the predicted position, the gate size, all matches within the gate, and the uncertainties on both the prediction and the matches.

The system was tested in simulation on several terrain types using a deep-ocean bathymetry map of the western flank of the Mid-Atlantic Ridge between the Kane and Atlantis Transforms. Results show more accurate navigation in the areas with greater bathymetric variability and less accurate navigation in flatter areas with more gentle terrain contours. In most places, the uncertainties assigned to the navigation positions reflect the ability of the system to follow the true track. In no case did the navigation diverge from the true track beyond the point of recovery.

Thesis Supervisor: W. Kenneth Stewart Jr., Ph.D.
Title: Associate Scientist, MIT/WHOI Joint Program

I returned, and saw under the sun, that the race is not to the swift, nor the battle to the strong, neither yet bread to the wise, nor yet riches to those of understanding, nor yet favor to those of skill, but time and chance happen to us all.

King Solomon
Ecclesiastes 9:11

Acknowledgments

So many have helped in so many ways, how do you say thank you? Listing the names just doesn't seem to be enough.....

Thanks to:

all the current and former members of the Deep Submergence Lab who helped keep the computers going and keeping track of who and what was where when and what was going on; Larry Flick, Beven Grant, Jon Howland, Steve Lerner, Marty Marra, Dan Potter, Craig Sayers, Cindy Sullivan.

all those at the WHOI Education Office for help, support, and giving me a place to live; Gary Bernston, Stella Callagee, Marcey Simon, MaryJane Tucci, Julia Westwater, and especially John Farrington and Jake Peirson for important conversations, advice, and action.

John Leonard and Rob Fricke for early discussions on this research.

present and former AOPE students John Buck, Erik Burian, Helen Huang, Phil Lebas, Todd Morrison, Hanu Singh, Tad Snow, and Xiaoou Tang for commiserating on the state of affairs and life in general.

Sabina Rataj for sticking through the ups and downs of 2 departments, 8 years, and several professors. Thanks for being a great friend!

good friends local and distant for good food, good fun, and making me smile; Drew Bennett, Ewann Bernston, Amy Black, Shelley Bouthillette, Eddie Chalom, Blair Cohen, The DeWilde and Flores families, Joice Himawan, Stefan Hussenoeder, Kirsten Laarkamp, Craig Lewis, Rulane Merz, Todd Morrison, Karen Palmer, John Pitrelli, Bonnie Ripley, Cindy Sellers, Dave Shull, Hanu Singh, Tad Snow, Miles Sundermeyer, Rebecca Thomas, Bill Williams, and Sarah Zimmermann.

my 35+ roommates who put up with ice hockey gear all over the living room.

the WHOI hockey team who let me and my hair scrunchies play WING!!!!, not laughing at me too much, and letting me whack the puck instead of something less appropriate.

Eric Grimson for being on the committee and supporting the big picture.

A special thanks to:

Al Bradley and Sandy Williams for encouragement, concern, insight and all the cheering (both kinds!). I appreciate your efforts!

Mark Grosenbaugh for stepping in when it counted and for chairing my defense. Thanks!

Henrik Schmidt for joining my committee and adding me to the long list of students he has rescued again and again and again and again..... Grazie Mille!!

my chairman Ken Stewart for his time, effort, input, insight, editing, and knowledge, and for making me a stronger person. Thanks!

my parents, brother, and extended family for all the love and support that 13 years at MIT requires.

The Good Lord for all the gifts I have been given.

Farewell MIT, I don't think I shall miss you.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

There are a variety of sensing platforms available for use in the ocean. These include surface ships, towed sleds, manned-submersibles, remotely operated vehicles, anchored or free-floating buoys, and bottom-mounted instrumentation. Each of these systems has its place in oceanographic research, but deployment, operation, and recovery can be expensive, and their usefulness can be limited in undesirable ways. For some applications, an autonomous underwater vehicle (AUV) could prove to be less costly in time, effort, and money, while at the same time providing better, more reliable data.

Autonomous underwater vehicles are robotic submersibles that have their own on-board computers and power supply. They are untethered, free-swimming vehicles capable of making and executing decisions, without any human interaction. AUVs are particularly suited for missions that involve underwater inspection, observation, surveying, monitoring, mapping, or in situ data analysis. They are also very useful in hard-to-reach places such as at great depths or under ice. There are over 50 AUVs that have been built and tested, dating back to 1963 [Nodland]. Surveys of present and former AUVs can be found in [Bellingham 1992b; Bergem; Busby].

This introduction presents some potential applications and then discusses some of the key technologies for enabling AUVs to do useful work. A critical technology is navigation, the focus of this thesis. This chapter establishes the research area in which my work contributes.

1.1 Applications of AUVs

To date, the number of successful and useful scientific or engineering missions completed by AUVs is very small, but the potential is very large. AUV missions fall into several categories: water column, seafloor, search, inspection, fishery related, and military [Bellingham 1992b]. Water column surveys collect data on temperature, conductivity, turbidity, turbulence, and other ocean properties. For example, AUVs can be used to track tracers (dyes) and monitor deep-water diffusion or water transport to aid physical oceanographic research. They can also monitor oxygen and pH levels in lakes, ponds, and estuaries, or explore areas with minimal access, such as water-filled volcanic craters. They can also evaluate the extent of dangers caused by chemical spills.

Seafloor survey missions can be geologically based and designed to image and map the bottom or to take rock, mineral, or sediment samples. AUVs can also be used to deploy and retrieve oceanographic instrumentation, monitor marine waste-disposal sites, and conduct fate-of-pollutant studies.

In search missions, AUVs can use visual or sonar techniques to seek out objects or locations of choice. Search missions also include detection and investigation of magnetic signatures for object location and identification. Wreck discovery and localization are also important search missions for historical, archeological, and salvage purposes. Disabled submarine location for rescue and recovery is also a search mission.

There are many possible applications for inspection and maintenance. AUVs might tend to underwater structures such as off-shore oil platforms, pipelines, cables, well-heads, tower bases, or dams, collecting information or making simple repairs. They could inspect ship hulls or water-filled areas of ships for structural integrity or to remove bio-fouling.

Maintenance of artificial reefs, sewage outfalls, or nuclear waste sites are other applications. Neutralizing toxic pollutants or rapid-response to potentially dangerous wrecks are also possibilities.

AUVs could aid the fishing industry by observing fishing gear in situ or assessing fish stocks. They can also be used to monitor fish behavior and track migratory patterns or identify potential new fishing sites.

Many military applications are possible for AUVs, such as independent patrolling for surveillance and data collection in sensitive areas. AUVs could be used for mine hunting, identification, and elimination, or as decoys in antisubmarine warfare or acoustic targets for military training and exercises. More offensively, they could lay mines or participate in cooperative attacks.

1.2 Key Technologies

The applications above are only some of the AUV missions which have been proposed, but to achieve these missions, several key technologies must first be addressed. These include the power source, the propulsion system, the high- and low-level control systems, and the navigation system. The mission specific sensors, such as a mapping sonar or temperature and conductivity probes, are additional technological requirements. Each of the core technologies is equally important to successful operation of the AUV.

The main limiting factor for AUV mission time is the power source. Currently, research is being conducted to evaluate and develop a variety of power sources such as fuel cells, seawater batteries, lithium batteries, etc. [Rao et al.; Walter; Willen]. There are also studies concerned with minimizing the amount of power drawn by a subsystem by

putting it to sleep when it cannot provide useful data or is not needed for a portion of the mission [Bradley].

The propulsion system is a critical component for the maneuverability of the vehicle. Some AUVs have the ability to hover, make sharp or stationary turns, or move independently in the vertical direction. For some missions, these capabilities are very important, but they tend to require a more elaborate propulsion system. Other missions, such as some surveys, do not require such difficult motions and can have simpler propulsion and maneuverability systems.

Considerable research has been done on both low- and high-level control systems [Turner; Uhrich and Walton; Watkinson et al.; Xu and Smith]. The high-level systems direct the mission by making larger decisions about goals and paths. This information is then passed to the low-level controller, which generates the specific commands to achieve the goals with the best possible performance. Different high- and low-level controllers are better suited for different vehicle designs and mission requirements.

This thesis is concerned solely with the navigation system. Navigation in general terms is the science of plotting, ascertaining, or directing the course of oneself or one's vessel. To successfully navigate from place to place, the three basic questions that must be answered are: Where am I?, Where am I going?, and How shall I get there? [Leonard; Leonard and Durrant-Whyte]. The first is the most fundamental localization question, and it is the purpose of this work to provide a method for answering it. The other two questions usually involve interaction between the high-level decision-making process and the low-level command-execution process. Some systems take a unified approach to navigation, where a high-level process integrates mission goals with path planning and obstacle-

avoidance processes to optimize one or several parameters, such as maximum coverage or minimum path. This thesis, however, does not take a unified approach to navigation and addresses only the localization question.

There are many different types of navigation being researched for use on AUVs. Some systems are based on techniques developed for terrestrial mobile robots and some are adaptations of techniques that have been used for surface ships. A few are discussed in Chapter 2, but the focus of this thesis is a bathymetric, terrain-relative positioning system.

Terrain-relative navigation has been used in several ways for vehicles traveling over land. One impressive application is the cruise missile, which updates its position periodically using stored maps of small areas between the launch and target sites. Bathymetric navigation has been discussed for surface vessels for over 25 years [Cohen], but it has yet to be exploited to full potential because of the lack of accurate maps of the ocean floor. In fact, only recently have we discovered that approximately 80% of the seafloor contains interesting bathymetric features, which may have enough variability to support bathymetric navigation [Dyer]. Of course, navigation will still be difficult in the flat areas of plateaus and the abyssal plains, but the abundance of seamounts, canyons, knolls, ridges, trenches, banks, rises, slopes, shoals, and troughs may provide sufficient relief for navigational purposes. Qualitatively, the greater the bathymetric variability and the finer the map resolution, the more distinct any piece of bathymetry will be. Consequently, the more accurate navigation is likely to be. This is discussed in more detail in later chapters.

Bathymetric navigation is dependent on the availability of a sufficiently accurate map during the mission, and much work is being done to provide them [Blackington et al.; de

Moustier and Matsumoto; Kamgar-Parsi et al.; Singh; Singh et al.; Stewart 1988; Stewart 1991]. In some cases, this means developing and improving techniques to process data taken by available sensors and to produce a map with an explicit knowledge and understanding of its accuracy. In other cases it means developing new sensors that can provide more accurate data from the start. This thesis does not address the issues of sensor design, sonar processing, or map building, but instead assumes that one of the available procedures has already provided a digital bathymetric map with known accuracy and resolution. The map may have a high level of uncertainty that will result in less precise positioning, but it is sufficient to have the map and to know what that uncertainty is.

The major contribution of this work is the development of a terrain-relative navigation system for AUVs that considers multiple location measurements at a single time step and explicitly represents the localization uncertainty at all times. The system incorporates data-association techniques that are usually used for target tracking and adapts them to this application. Image-processing algorithms are used to obtain the map-matched position measurements and to enable the necessary calculations to be completed without prohibitive computational demands. The navigation system is scale-independent and will operate successfully using maps with a wide range of resolutions. Also, the system uses the uncertainty on the map in the localization algorithms, so that highly uncertain maps or maps with varying degrees of uncertainty can be used. The system can be used for any application discussed in Section 1.1, provided that a bathymetric map is available and that the map resolution is sufficiently fine to support the navigation accuracy needed for the specific mission.

1.3 Document Organization

Chapter 2 discusses a variety of research in the field of underwater vehicle navigation. Presented is a brief summary of navigation systems that have been suggested, researched, and tested for AUVs. This chapter also introduces terrain-relative or bathymetric navigation by citing related work, particularly that on the cruise missile and at the Norwegian Defense Research Establishment.

Chapter 3 presents the two basic algorithms of the terrain-relative navigation system. The matching algorithm compares the measured bathymetric profile to the supplied map for a defined search area. Matches are ranked by the mean absolute difference (MAD). An intelligent, adaptive-search method using a coarse-to-fine algorithm with a modified beam search is employed to reduce the computational requirements for determining good matches. The updating algorithm, grounded in the Kalman Filter framework, consists of a Probabilistic Data Association Filter with Amplitude Information (PDAFAI) that weights each accepted match according to the value of the MAD and the size of the innovation. A best estimate with uncertainty (covariance matrix) is calculated for each time step.

Chapter 4 discusses the interactions between the matching and updating processes presented in Chapter 3. The navigation uncertainty is used to determine the search range, and the error values of the accepted matches are used to calculate the probabilities of detection and false alarm which affect the match weighting. The measurement covariance on the matches affects both the position estimate and covariance and is determined a priori by direct matching of noiseless data to the available terrain map.

Chapter 5 presents simulation results for five navigation tracklines across varied terrain of deep-ocean bathymetry. Results are presented both in pixel coordinates, to

emphasize the scale-independent nature of the system, and in actual dimensions (meters), to quantify success numerically with a physical parameter. The results vary depending on the terrain in which the profiles were taken; however, the system never lost track and the uncertainties were always bounded.

Chapter 6 gives the conclusions drawn by the data produced for Chapter 5. Methods for improving the terrain-relative navigation system are discussed, along with further directions for research in this area.

Chapter 2

Underwater Vehicle Navigation

Many navigation technologies are possible for use on an AUV. Some require accurate knowledge of the vehicle state, and others are dependent on a reliable means of communication underwater. Two broad categories of navigation are internal and external systems. Internal navigation systems do not measure position, but rather determine location internally by integrating velocities and/or accelerations. External systems determine positions relative to a property or feature of the environment. This chapter categorizes and discusses many navigation methodologies that have been used for AUVs and introduces related work in terrain-relative navigation.

2.1 Internal Navigation

Internal systems navigate using relative information only. They require accurate knowledge of the vehicle state and depend on sensors to provide measurements of the derivatives of the state. Examples of internal navigation are dead reckoning and inertial navigation.

2.1.1 Dead Reckoning Navigation

Dead reckoning is the most fundamental of all navigation systems and requires the least amount of information. It simply updates the position by using a measured velocity vector (magnitude and direction). Unfortunately, the error associated with dead reckoning

increases with time, and there is no built-in method for reducing this error. Depending on the sensors used and the specific vehicle mission, the navigational error can grow rapidly to the point where either the mission will not produce useful data or it will not be achievable at all.

Recently, the cost and size of Doppler velocimeters have been reduced to a level where integration into an AUV has become more practical. Dopplers can provide very accurate velocity measurements under the proper conditions, but still have drift rates that preclude good navigation over time. Relative-position errors are typically on the order of 1% of the distance traveled [Bellingham 1992a]. Another problem that can occur with Dopplers is the loss of bottom track.

2.1.2 Inertial Navigation

Inertial navigation systems are also in the internal navigation classification, but instead of velocities, accelerations are measured and integrated twice to give position. Generally, inertial systems produce more accurate navigation than dead-reckoned systems. As with Acoustic Doppler Current Profilers (ADCPs), the size and cost of inertial measuring packages have recently been reduced so that their use on an AUV is more practical. Drift rates of medium- and high-accuracy inertial navigation systems are roughly 1 nmi/hr and 0.25 nmi/hr, respectively [Foxwell and Hewish].

2.1.3 Recursive Navigation

Typically a Kalman Filter is used in all types of navigation so that a best estimate of position can be made while accounting for and explicitly representing the uncertainties on both the state estimate and the measurements. [Leader] developed an internal navigation

system that uses measurements from both an ADCP and an inertial measurement unit to achieve localization with an Extended Kalman Filter. He shows that if the filter is designed properly, to account for real-life complications, reasonably successful navigation can be obtained for a vehicle that is not maneuvering in a complicated fashion.

The Kalman Filter framework is actually a quite common technique in both terrestrial mobile robot and AUV navigation, because a vehicle will typically use both internal and external navigation systems. The internal system provides the predicted location by integrating measurements of other states, and the external system provides a direct measurement of location. Chapter 3 presents the Kalman Filter in detail.

2.2 External Navigation

External navigation requires knowledge of, or the measurement of, something external to the vehicle. Included in this larger category are electromagnetic, optical, acoustic, and geophysical navigation systems.

2.2.1 Radio Navigation

Surface vessels make extensive use of Global Positioning Systems (GPS or Differential GPS), which depend on the receipt of electromagnetic radiation. Unfortunately, radio waves are attenuated quickly in seawater and can only be used at very close ranges. It is possible for the AUV to surface periodically to obtain a GPS fix, as naval submarines do, but there are several disadvantages to this [Kwak et. al]. For an AUV to use a form of radio navigation, it must interrupt its mission, expend time and energy surfacing, risk its safety for up to a minute on the surface getting the fix (which is

especially dangerous if near shore or in a shipping channel), then expend additional time and energy submerging to resume the mission. Even if an extremely accurate fix is obtained, the vehicle location uncertainty can grow significantly during descent before the mission is ever resumed.

2.2.2 Optical Navigation

Optical systems have transmission-loss problems similar to those of electromagnetic systems [Chantler et al.]. Navigation is lost as soon as visibility of the surrounding environment becomes poor, which usually happens at very short ranges. With optical systems, the AUV must carry an on-board light source, associated electronics, and the power to generate sufficient light. These power requirements can be substantial even for close ranges, as so much of the light is dispersed, absorbed, or reflected away from the video device. Use of a blue or green laser can increase the altitude, but for most applications, this distance is still unacceptably small. Nevertheless, research is still being done in the area of determining relative motion by using a sequence of images taken by a CCD camera [Aguirre et al.].

2.2.3 Acoustic Navigation

Acoustic navigation is the most widely accepted form of AUV navigation, and a variety of systems have been both researched and tested. Most require an engineered environment, meaning that something has been added to the environment to aid navigation [Gat]. Usually, the engineering takes the form of one or more acoustic beacons. Although the principles behind each of the acoustic navigation schemes is quite different, they are all susceptible to the same general sonar problems: multipath, signal refraction,

reverberation, 3D spatial variation of the sound speed, shipping noise, biological noise, vehicle self noise, wind/water interaction, and sensor accuracies dependent on hydrophone spacing and source frequencies [Calcar].

2.2.3.1 Short-Baseline Navigation

Short- (SBL) and ultra-short-baseline (USBL) systems are used to determine the direction of propagation of the acoustic signal and therefore the direction vector to a single beacon. Short-baseline navigation uses multiple hydrophones mounted on the AUV with moderate separation and measures the differential time of arrival of a single pulse. Ultra-short-baseline navigation systems [Chen and Alvarado; Tracey] use hydrophones with very small separation and measure the phase difference of the carrier pulse at two or more locations. Ranges to the single beacon are calculated by one- or two-way travel times, depending on clock synchronization and whether or not the vehicle has a source.

2.2.3.2 Long-Baseline Navigation

Long-baseline (LBL) navigation systems use an array of acoustic beacons separated by a range of 100 m to a few kilometers [Bernsten; Calcar; Di Massa; Geyer et al.]. The vehicle determines its position by listening to the pulses emitted from the beacons and recording the arrival times. The locations of these beacons must be provided, and the vehicle must be able to detect and distinguish between their signals. Currently, long-baseline navigation is the most widely used technique for AUV localization.

The two major types of long-baseline navigation are spherical and hyperbolic. In spherical navigation, the vehicle interrogates the array by emitting its own pulse then listens for the responses from the beacons. By measuring the elapsed time and knowing

the sound-speed profile, the range to each beacon can be determined. Each range measurement confines the vehicle to be on the surface of a sphere centered at the beacon and with radius equal to the range. The point at which all spheres intersect is the vehicle position. Very accurate navigation $O(<1\ m)$ is possible with this system, but it does require the deployment and calibration of the beacon array. Some commercially available long-baseline navigation systems [Marquest], which use higher bandwidth, claim centimeter accuracy, but this is only valid in the best of conditions.

In hyperbolic long-baseline navigation [Bellingham et al. 1992; Di Massa], the vehicle does not interrogate the array, but instead listens passively to the synchronized pulses emitted by the beacons. Hyperbolic acoustic navigation is patterned after LORAN, a radio navigation system developed during World War II by the MIT Radiation Laboratory [Pierce; Pierce et al.]. Unlike spherical navigation, hyperbolic navigation measures the differential arrival times of the beacon signals to constrain the vehicle location to be on hyperboloids instead of spheres. Often, a depth measurement is added to aid in resolution. An advantage of using hyperbolic acoustic navigation is that multiple vehicles can operate at the same time without interference. Another advantage is that hardware and power requirements on the vehicle are reduced because the vehicle does not interrogate the array. Also, somewhat clandestine operations can be undertaken because the vehicle listens only and does not give away its own position. However, hyperbolic navigation is usually less accurate $O(3-5\ m)$ than spherical navigation, and the accuracy is highly dependent on the location within the array. [Deffenbaugh 1997] compares the accuracies of these systems in greater detail.

A major disadvantage of long-baseline acoustic navigation is that the calibration of the array, which must have accuracy better than the desired vehicle location accuracy, can be a major undertaking, both initially and during periodic maintenance. The cost in time, equipment, and effort used can be expensive, especially if many beacons are used in a fairly deep area far from shore. The vehicle also has a reduced level of autonomy because it must stay within the array. Another important issue that can arise is the loss of navigation in shadow areas, regions acoustically unreachable due to the local topography.

Over ranges greater than a few kilometers, variations in the sound speed can complicate the relationship between range and travel time, which leads to reduced accuracy of most long-baseline navigation systems. [Deffenbaugh 1994] has explored the possibility of using a ray-tracing model of sound-speed propagation for long-range AUV navigation. His method associates signal arrivals with specific ray paths, and path inversion provides vehicle position. This is especially useful in high-multipath environments, such as near bottom or under ice. Preliminary tests with this system using multipath matching has produced positional errors of a few meters for ranges up to 5 km. However, there is an important practical consideration that limits the implementation of this system. It is difficult to mount on an AUV a receiving array that measures ray arrival angles accurately enough to produce good navigation.

2.2.4 Geophysical Navigation

Recently there has been an increased interest in geophysical navigation systems that use gravity, magnetics, and bathymetry as navigational aids. In most cases this requires a priori maps of these characteristics. Consequently, it is appropriate to discuss maps before

proceeding. However, although map-building is a very important and broad area of research [Ayache and Faugeras; Connell; Elfes; Engleson and McDermott; Hebert; Kuipers et al.; Langer and Hebert; Singh; Stewart 1988], it is not the purpose of this research to construct maps, so this discussion is brief.

When using a map, the features or characteristics within should be directly related to the purpose for which the map is created. The map detail should also be dictated by the intended application as well as by the accuracy of the available measurements, the storage requirements, and the time available for data manipulation [Cox and Wilfong]. Generally, map representations are either continuous or feature-based. Continuous representations have values of the parameters of interest available or calculable at all times. These include splines defined by the parameter values [Dierckx; Guibas and Ramshaw; Tuohy] or grid-based representations where there is a parameter value for each point of the grid and interpolation between the grid points is possible [Elfes; Hunter and Steiglitz; Matthies and Elfes; Moravec and Blackwell]. Feature-based representations identify important features, landmarks, or characteristics of the environment [Kuipers and Byun]. These can be extremal points, such as minima or maxima [Orser and Roche], or physical features such as walls or intersections [Drumheller; Leonard].

When using maps for navigation, it is common to engineer or seed the environment with navigational aids, such as the acoustic beacons mentioned above. These aids become features. However, for geophysical navigation, one could engineer the environment by placing objects within it or exploit its natural structure and use a continuous representation.

2.2.4.1 Gravitational Navigation

The earth's gravitational field varies with its surface structure, density, and thickness, and it is theoretically possible to use either gravity variations or anomalies for navigation. [Jircitano et al.] state that gravity-based systems have been successfully used on land, aircraft, and sea surface vessels. They also present simulation results for AUVs and predict success if a quality gravity gradiometer is used. However, it is not yet clear whether bounded long-term autonomous navigation is possible.

2.2.4.2 Magnetic Navigation

Geomagnetic position may be one of nature's best navigational schemes, as some research suggests that birds, fish, and other animals use magnetic intensity as a navigation aid [Waterman]. Magnetic anomalies occur naturally in the earth's crust but can also be generated by man-made objects; both produce 3D spatial magnetic variations, [Parker and Heustis; Polvani; Tuohy; Tyren 1982; Tyren 1987]. The field variations can be detected passively to enable a variety of missions including pipeline following and mine hunting. Essentially, the minimum and maximum values of the magnetic anomalies can be viewed as magnetic beacons. [Foxwell and Hewish] also indicate that pre-positioned magnetic members may enable the vehicle to reset its navigational uncertainty and assist in maintaining bounded navigational errors.

2.2.4.3 Bathymetric Navigation

The third geophysical parameter of interest is the bathymetry. Although terrain variations have been exploited for land-based vehicles and aircraft, little has been done to exploit sub-surface terrain variations. As with magnetics and gravity, this has primarily

been attributed to the lack of accurate maps. If there is enough local variation in the seafloor, navigation relative to the bathymetric contours or to identified features should be possible. Section 2.3 presents terrain-relative navigation using bathymetric contours, which is the focus of the research in this thesis. To date, it appears that true feature-based navigation has not been attempted for AUVs.

2.2.4.4 Hybrid Geophysical Navigation

[Tuohy] shows that multiple geophysical properties can be used to assist navigation. Using interval B-splines, he presents an algorithm for navigation based on the representation, abstraction, and interrogation of multiple geophysical maps. His demonstrations use bathymetry and magnetics, but are equally applicable to gravity. Tuohy uses an interval model to provide positional estimates with uncertainty based on where the vehicle is not, because when the impossible is eliminated, what remains must be the truth. His system provides navigation accuracy on the order of the accuracy of the supplied maps.

2.3 Terrain-Relative Navigation

The research of this thesis is confined to terrain-relative navigation based on natural topographic contours. The three most closely related examples of prior work are discussed below. The first is the cruise missile for which terrain navigation is the key to its capability. The second attempts to apply the cruise-missile techniques to AUVs. The third is the most thorough discussion of bathymetric navigation for AUVs to date.

A cruise missile is a pilotless, continuously self-powered, and self-guided vessel [Tsipis]. For the majority of its path, a cruise missile uses inertial navigation, but because the distance between launch and target sites can be large, the drift rates of the inertial navigation system can cause the position accuracy to grow too large for successful target acquisition. However, if the missile can periodically recognize its true position and compare this with its preassigned trajectory, then it can make the appropriate adjustments and bring itself back to the correct track. Furthermore, the error between the actual and intended position can be used to calibrate and to reset the inertial navigation system, potentially reducing future errors. To periodically calculate the actual position, a terrain contour matching method known as TERCOM is used.

TERCOM was conceived and patented in 1958, and [Golden] gives the history and development of the technique from 1958-1980. The system operates on the principle that "any single geographic location on the land surface of the earth is uniquely defined by the vertical contours of the surrounding terrain" [Golden]. The missile uses a downward-looking radar altimeter to obtain a sequence of terrain elevations along the flight path (see Figure 2.1). The guidance software then matches this measured profile with the stored local map and determines the geographic location of the measured sequence. The mean absolute difference is the quantity of interest for this comparison, and the match providing the minimum value of this parameter is selected. The navigation fix is then used to update the missile position and to adjust parameters of the inertial measuring unit.

The TERCOM system uses several maps between the launch and target sites, and each map provides one navigation update, as shown in Figure 2.2. Since the missile trajectory between TERCOM maps need not be a straight line, flight paths that avoid enemy
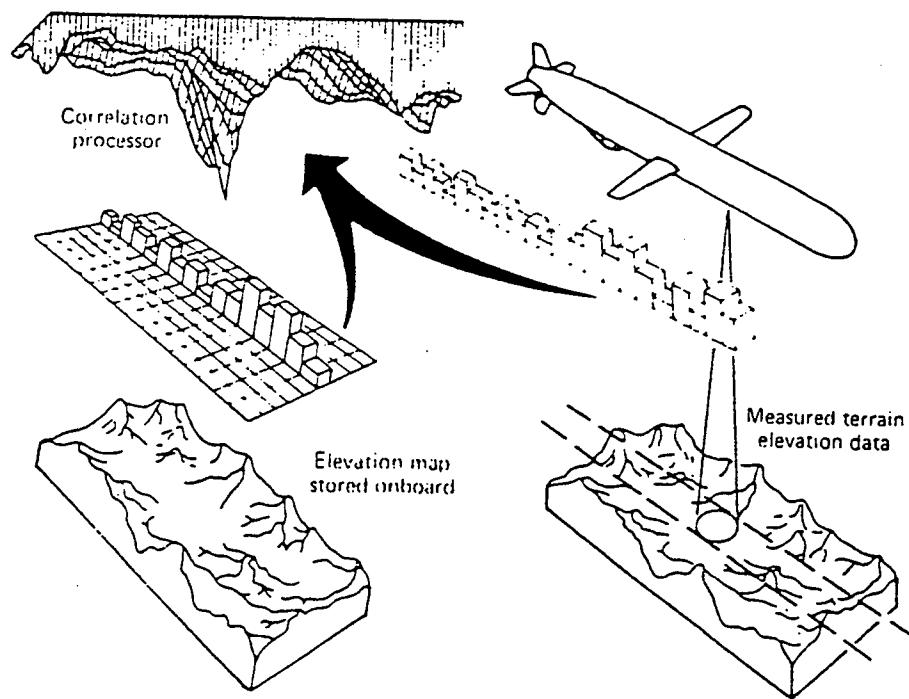
Figure 2.1. TERCOM Matching Process, from [Golden].
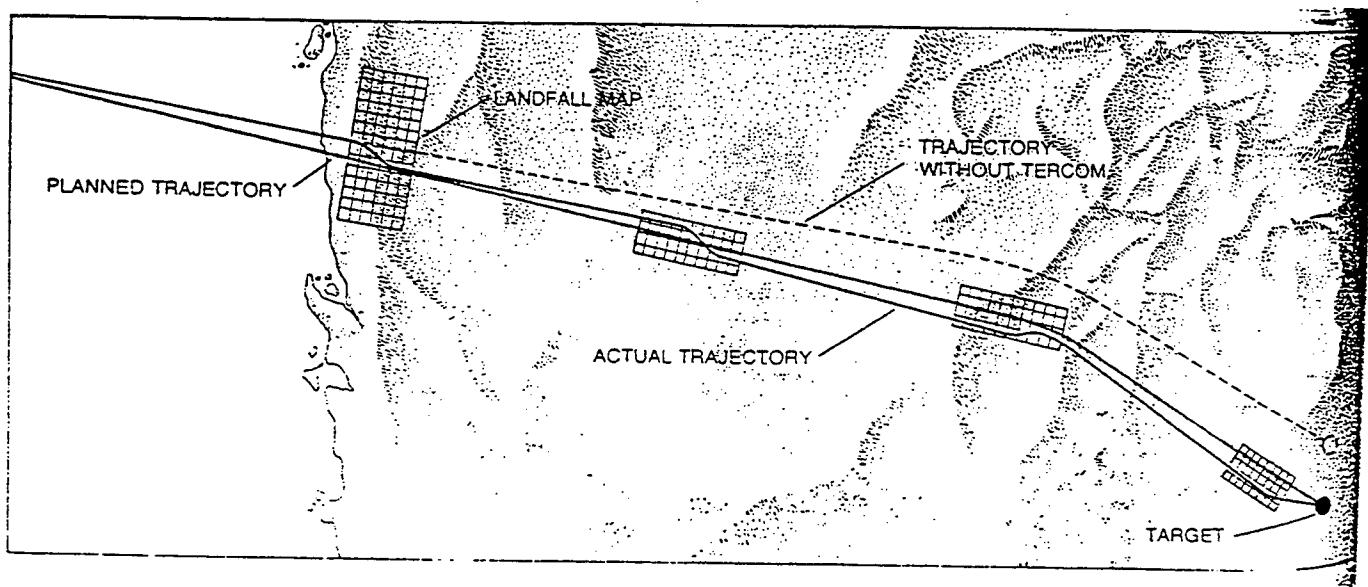
Figure 2.2. Overall TERCOM System Uses Multiple Maps, from [Tsipis].

defenses are easily accommodated. The biggest errors of the system are all map-related and are either due to map errors, poor choice of terrain, or the absence of suitable terrain to support the matching process. Good terrain must be more than just rough, it must be unique. With a quality map, navigation accuracy is comparable to the pixel size of the digital map.

[Kullander] investigated the feasibility of this technique for AUVs. In his work, he assumes that the AUV has an on-board inertial navigation system and uses terrain navigation to periodically update the position. He employs a global map with coarse-scale data and patches of fine-scale data, which he calls correlation areas. He concludes from simulation that applying this technique to underwater environments is much more difficult than on land because it is more difficult to collect the necessary data and, furthermore, that data may have large errors. The success of the operation may also be dependent on knowledge of the properties of the water, which affect the performance of the sensor.

[Bergem] takes the terrain contour matching concepts one step further and provides the first thorough evaluation of terrain-relative navigation for AUVs. In his work, he investigates the feasibility of using data from a multibeam sonar and matching the profiles to a pre-stored map continuously. He does not have patches of fine-detailed maps but assumes he has sufficient detail everywhere, so the entire region can support bathymetric navigation. His system is built around a Kalman Filter and was implemented and tested with real data from the Oslo fjord of Norway. He successfully demonstrates that navigation with bounded errors can be achieved solely based on information from bottom topography. His research also shows that for the data set he used, "it seems that the position errors are mainly caused by errors in the measurements or in the reference map

and not by a random error caused by too little variation in the terrain." Although true for experiments in the Oslo fjord, this may not be true in general for all terrains.

In Bergem's system, he uses the navigation uncertainty to define the search range in which he compares the profile to the map. When this search range is large, the extensive computations necessary to test every location become time consuming. He does not address this issue.

Bergem also presents briefly the concepts of data association, i.e., the methods of choosing and weighting appropriate measurements. Although he selects only one match, he emphasizes directly the importance of using all information available. He takes only the best match for simplicity, because data-association techniques were "originally designed for target tracking, and the parameters that are needed are difficult to estimate for other applications, e.g., the target detection probability and the spatial density of false alarms." Despite this problem, data-association methods have many of the required characteristics needed for a navigation system, and although they are grounded in the Kalman Filter framework, they are non-linear and more complex than a standard Kalman Filter.

The motivations for my research are similar to Bergem's. First, techniques using terrain variability have proved successful for land-based operations, and it is now known that the seafloor has sufficient topography to support navigation in the same fashion. Because bathymetric sonars have improved significantly lately and will continue to do so, there has been an increase in the development and improvement of bathymetric maps. Furthermore, a bathymetric sonar may already be on-board the AUV, for mission requirements. With terrain-relative navigation, the vehicle is neither dependent upon or

limited by the number and positions of deployed beacons. Extra time, money, and effort need not be spent designing, deploying, calibrating, and maintaining an acoustic beacon array. The navigational accuracy needed for the AUV mission will dictate the accuracy and the spacing needed on the supplied map.

For my research, I address the same issues as Bergem; however, I am concerned about the processing time required. Computer processors are becoming faster and more powerful as research in that field continues, so someday it may be possible to process the large volume of information needed for this technique in a very short period of time. However, for now and the immediate future, an intelligent rather than an exhaustive search seems prudent.

I also agree that it is wiser to use multiple pieces of information gained by matching the profile to the map, rather than by choosing only the best match. In many cases, the uncertainties involved can result in the best match not providing the best navigational update. I do use the complex algorithms and techniques of data association and do encounter the difficulties predicted by Bergem, but I have found realistic solutions for this application. The following two chapters explain my technique in greater detail.

Chapter 3

Terrain-Relative Navigation: Principal Algorithms

Terrain-relative navigation is a map-matching methodology that assumes the vehicle has both a digital bathymetric map stored on-board and the ability to image the seafloor. The crux of the navigation problem is to localize the sonar image in an existing global map. Suppose there is enough information to determine a search area in the map within which the image should be localized. Ideally, the task at hand is to find the best match between the image and the map for all possible positions and orientations in this search area. However, nothing is perfect, there is always uncertainty. To cite only a few sources of uncertainty in the sonar image, there is uncertainty due to noise on the sonar returns, uncertainty due to the accuracy of the sonar, and uncertainty in the sonar processing. There is also uncertainty on the supplied map. When all of the sources of uncertainty are considered, one must conclude that simply finding the best match between the sonar image and the map may not result in the best localization. It is also evident that it is important to explicitly represent the uncertainty on the accepted vehicle location. However, the description of terrain-relative navigation begins under the assumption that the best match is what is sought. A second reason to be suspicious of the best match will soon be discussed

This chapter presents the two principal algorithms of terrain-relative navigation, namely, the matching algorithm, which localizes the image in the map, and the updating algorithm, which uses the matching information to generate navigation positions.

3.1 Matching Algorithm

To find the best match, first, a criterion for determining the "goodness" of a match (the agreement between the image and the map) must be defined, and, second, this "goodness" must be evaluated throughout the search area. There are many resemblance coefficients that can be used to represent "goodness." Some are similarity parameters, such as the Cosine Coefficient and Correlation Coefficient (also known as the Pearson Product-Moment), and some are dissimilarity parameters, for example the Canberra Metric and the Bray-Curtis Coefficient [Romesburg]. Each resemblance coefficient is a different metric that measures the degree of similarity or dissimilarity but may or may not indicate identity. [Lim]states that for digital signals, although there are many options, the "two most commonly used choices are the squared difference and the absolute difference." The goodness criterion chosen here is the mean absolute difference (MAD) between the bathymetry values of the image and the map. The absolute difference, and not the difference squared, was chosen to avoid magnifying error values due to fliers.

3.1.1 Coarse-to-Fine Search Method

Since it is entirely possible that the search area is large, an exhaustive search may prove to be computationally too costly and time-consuming, so an alternative approach must be found. A coarse-to-fine method is used in which both the map and the image are low-pass filtered and downsampled (say by 2), then matching is attempted at this downsampled or coarse level [Jolion and Montanvert]. Each selected match is then tracked back to the original resolution, and the corresponding match-location pixel is determined. Next, this pixel and all its adjacent pixels, diagonals included, are searched in the original

map. For example, suppose we have a 2D signal and the best match is at location (17, 6) in a 20 x 20 grid, (see Figure 3.1). Instead of searching all 400 pixels to find (17, 6), we low-pass filter, downsample by two, and search the downsampled area — only 100 pixels. The best match is found at location (9, 3), which in the original scale is (18, 6). Now, the patch (17:19, 5:7) of the original grid is searched and the best match is found at (17, 6). Note that with 2D signals, a match at the coarse level produces nine test points in the fine level.
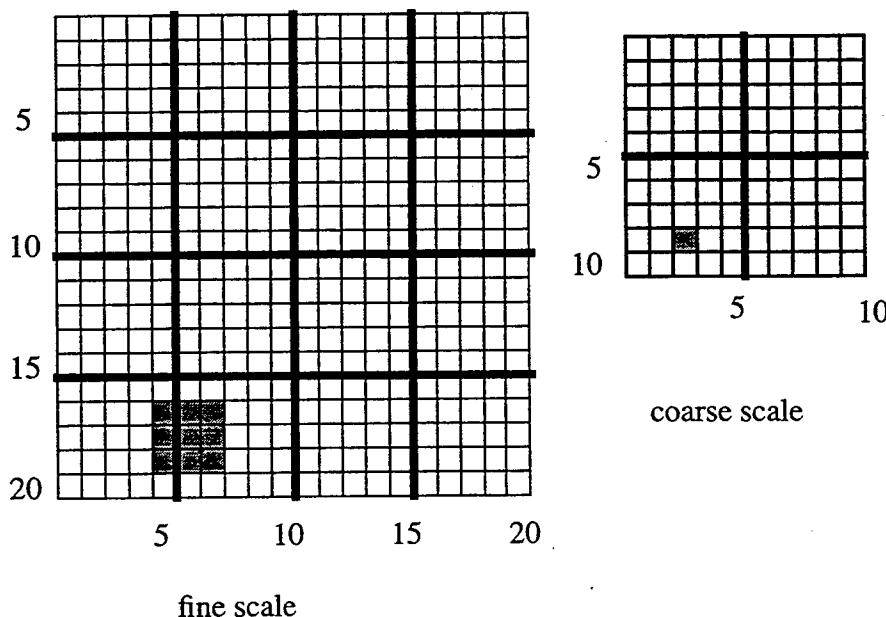
Figure 3.1. Coarse-to-Fine Example.
Pixel (9,3) on coarse scale matches to patch (17:19,5:7) on fine scale.

For large search areas the process is to use several coarse maps, each determined by downsampling by a different power of two, and to proceed in successively finer steps from the coarsest map to the finest map. For example, say the coarsest map and image are determined by low-pass filtering and downsampling by four, and an intermediate coarse map and image are determined by low-pass filtering and downsampling by two. We start at the coarsest map, find the best match, move to the intermediate coarse map, test the nine

41

options, find the best match, move to the fine map, test the nine options, and find the best match. In my algorithm the amount of downsampling at the coarsest level adapts to the size of the search range. For very small search ranges, no downsampling is needed, for moderately small search ranges, downsample by two, for moderately large to large ranges, by four, and for very large ranges, by eight. The boundaries between these four categories have been selected based primarily on simulation run-time expectancy (see Section 4.1).

Although the coarse-to-fine method usually reduces the computational cost sufficiently, there is an important drawback: *There is no guarantee that the best match produced by the coarse-to-fine algorithm will be the same as that found by doing an exhaustive search at the original scale.* Or, stated another way, *there is a probability that this algorithm will not 'detect' the true match* (see Sections 3.2.2.3 and 4.2). The low-pass filtering removes the high-frequency information in the signal, and downsampling reduces the total amount of information. This is the second reason why the best match should not simply be trusted to be the "right" match.

Instead of selecting a single match, a list of good matches is compiled. The number of entries in this list can be determined in several ways. Two options are: 1) to select the number arbitrarily and 2) to accept all matches that have a certain relationship to the best match. Two possible relationship algorithms are: 1) threshold option — accept the matches whose MAD is no greater than a certain value over the MAD of the best match and 2) percentage option — accept the matches whose MAD is within a certain percentage of the MAD of the best match. The relationship criterion with the percentage option is selected because it maintains the most flexibility.

3.1.2 Beam Search

To merge the concepts of a set of matches with the coarse-to-fine matching algorithm, the scale space is considered to be a tree with the coarsest level as the top level and the finest level (original resolution) as the bottom level. A modified beam search is used to track the matches through the tree from top to bottom. In a beam search, an arbitrary number w of matches is selected and the best w matches at the highest level are chosen [Winston]. All branches from these w nodes are followed to the next level. Suppose that each node has three branches so there are now $3w$ nodes to consider (assuming the tree is simple). These $3w$ nodes are then evaluated and the top w of these nodes are selected. The branches from this new set of w nodes are followed to the next level, and the process is repeated until the lowest level of the tree is reached. So, at each level there are always w nodes. This is *not* the same as simply tracking the original w nodes individually through the tree, because it is possible that several of the selected nodes may be branches of the same parent node one level above. Figure 3.2 shows a simple example for the case where w=3 and each node spawns three branches.
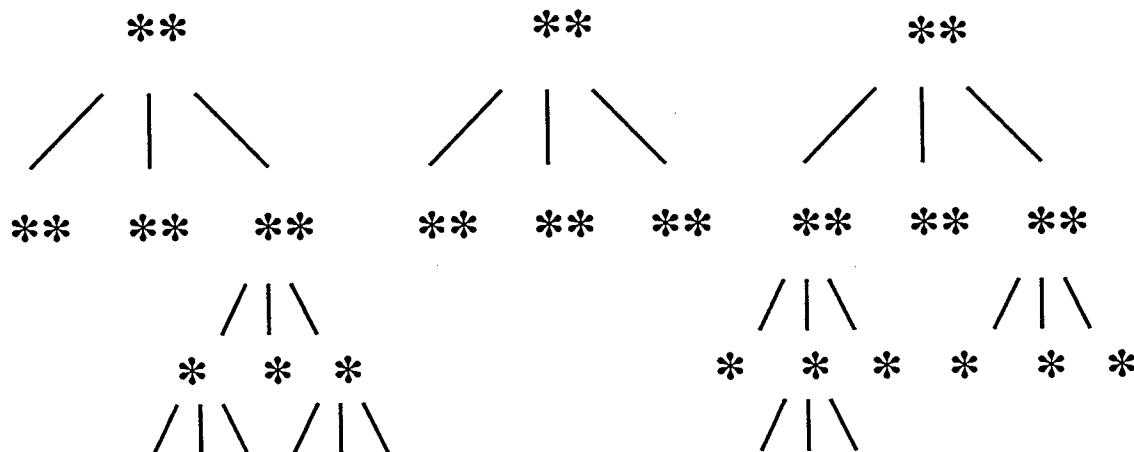
Figure 3.2. Beam-Search Tree with w=3.
Coarsest level at top; finest level at bottom.

Two issues in my application make the beam search slightly more complicated than that just described. First, the $9w$ test nodes (nine because there are two dimensions) at each level may not be independent, as my tree is not simple. For example, suppose we return to the previous case of the 20 x 20 grid. Suppose also that $w=2$ and the best matches at the coarse level are (9, 3) and (8, 3), (see Figure 3.3). Converting these to the original scale,

coarse scale
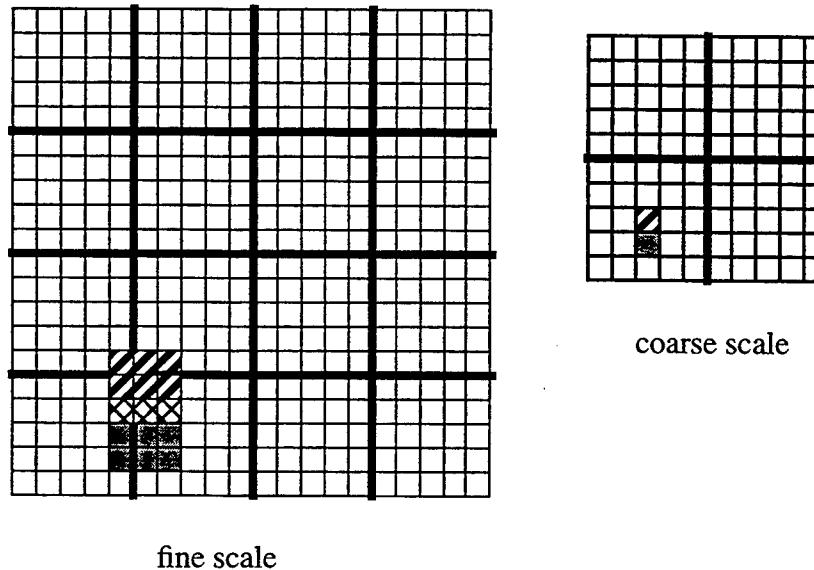
fine scale

Figure 3.3. Coarse-to-Fine with Beam Search Example.
Pixels (8,3) and (9,3) on coarse scale match to patches (15:17,5:7) and (17:19,5:7)on fine scale. Patches overlap on (17,5:7).

the two patches to be searched are (17:19, 5:7) and (15:17, 5:7). The test nodes (17, 5), (17, 6), and (17, 7) appear twice. Since it is desirable to test each node only once, the list of test nodes is reduced so there are no duplicates, and consequently there are less than $9w$ nodes to test.

The second issue is that since the percentage option is used for selecting the set, the number of nodes w at each level is not constant but entirely dependent on the error values of each measurement relative to the error of the best match. It is conceivable that w could

44

be a large number at a certain level of the tree and then a small number (even one, but never zero) at the next level of the tree.

3.2 Updating Algorithm

The matching algorithm provides a set of possible match locations, but to complete the localization problem the information in these matches must be combined to determine a best estimate of the true location. I turn to estimation theory to achieve this. To keep terminology consistent with that field, each match in the list is designated a measurement and the true state is the true location of the image in the map. The framework used in this estimation problem is established by first outlining the Kalman Filter then describing the data-association techniques that build upon this framework to achieve the actual updates. Much of the information presented below is well-established in the literature, and it is not my purpose to analyze that work. However, because of its importance to the task at hand, the fundamental procedures are described here.

3.2.1 Kalman Filter

The Kalman Filter is a recursive least-squares estimation technique that uses a Bayesian approach [Bar-Shalom and Fortmann; Brown and Hwang; Gelb]. The discrete linear Kalman Filter is outlined here, without derivation, using the notation in [Bar-Shalom and Fortmann].

The state equation for state x at time step $k+1$ is

$$x(k+1) = F(k)x(k) + G(k)u(k) + v_N(k) \quad , \tag{3.1}$$

where $x(k)$ is the state at time step k, $F(k)$ is the transition matrix containing information about how the state evolves if no noise or inputs are made to the system, $u(k)$ is a known controlled input with associated gain matrix $G(k)$, and $v_N(k)$ is a sequence of zero-mean, white, Gaussian process noise with covariance matrix $Q(k)$. The measurement observation equation is

$$z(k + 1) = H(k + 1)x(k) + w(k + 1), \qquad (3.2)$$

where $z(k+1)$ is the measurement at time step $k+1$, $H(k+1)$ is the conversion matrix between the state and measurement assuming no noise, and $w(k+1)$ is a sequence of zero-mean, white, Gaussian measurement noise with covariance matrix $R(k+1)$.

The two equations above ideally represent the evolution of the true system and what the associated measurement vector should be. However, when operating in the real world, one can never know *exactly* what the state and noise are, so the above state equation cannot be used. Instead, the state prediction equation, which acknowledges that we can only proceed with an estimate of the state, is substituted. The predicted value of the state $\hat{x}$ at time step $k+1$ given all information up to and including that at time step k is

$$\hat{x}(k + 1|k) = F(k)\hat{x}(k|k) + G(k)u(k), \qquad (3.3)$$

where $\hat{x}(k|k)$ is the estimated state at time step k given all information up to and including that at time step k. The state prediction covariance matrix P at time step $k+1$ given all information up to and including that at time step k, which represents the level of error associated with the state estimate, is

$$P(k + 1|k) = F(k)P(k|k)F'(k) + Q(k), \qquad (3.4)$$

where the prime notation indicates transpose, and *P(k|k)* is the state covariance matrix at time step *k* given all information up to and including that at time step *k*. It can also be defined as

$$P(k|k) = E\left\{ [x(k) - \hat{x}(k|k)][x(k) - \hat{x}(k|k)]' | Z^k \right\}, \tag{3.5}$$

where Z^k represents all measurements up to and including those taken at time step *k*.

The predicted measurement is

$$\hat{z}(k+1|k) = H(k+1)\hat{x}(k+1|k), \tag{3.6}$$

and the resulting innovation *v*, defined as the difference between the actual and predicted measurements, is

$$v(k+1) = z(k+1) - \hat{z}(k+1|k). \tag{3.7}$$

The innovation covariance *S* is expressed as

$$S(k+1) = H(k+1)P(k+1|k)H'(k+1) + R(k+1). \tag{3.8}$$

Finally, the updated state estimate and state covariance at time step *k+1* are

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + W(k+1)v(k+1) \tag{3.9}$$

and

$$P(k+1|k+1) = P(k+1|k) - W(k+1)S(k+1)W'(k+1), \tag{3.10}$$

where the Kalman Gain *W* at time step *k+1* is

$$W(k+1) = P(k+1|k)H'(k+1)S^{-1}(k+1). \tag{3.11}$$

The one-cycle procedure outlined above is summarized by Figure 3.4 (derivations of these

equations are found in [Bar-Shalom and Fortmann; Brown and Hwang; Gelb]).

Evolution of True System | Controller | Estimation of State | State Covariance Computation

State x(k)

Control u(k)

State Estimate $\hat{x}(k|k)$

State Error Covariance P(k|k)

Transition
x(k+1)=F(k)x(k)+G(k)u(k)+v(k)

State Prediction
$\hat{x}(k+1|k)=$
$F(k)\hat{x}(k|k)$
$+G(k)u(k)$

State Prediction Covariance
P(k+1|k)=
F(k)P(k|k)F'(k)+Q(k)

Measurement Prediction
$\hat{z}(k+1|k)=$
$H(k+1)\hat{x}(k+1|k)$

Innovation Covariance
S(k+1)=
H(k+1)P(k+1|k)H'(k+1)
+R(k+1)

Measurement
z(k+1)=H(k+1)x(k+1)+w(k+1)

Innovation
$v(k+1)=z(k+1)-$
$\hat{z}(k+1|k)$

Filter Gain
W(k+1)=
P(k+1|k)H'(k+1)S^{-1}(k+1)

Updated State Estimate
$\hat{x}(k+1|k+1)=$
$\hat{x}(k+1|k)+$
$W(k+1)v(k+1)$

Updated State Covariance
P(k+1|k+1)= P(k+1|k)
-W(k+1)S(k+1)W'(k+1)

Figure 3.4. Kalman Filter State Estimation, one cycle, from [Bar-Shalom and Fortmann].

The Kalman Filter is the optimal estimator if all noises on the system and the

uncertainty of the initial state can be modeled by zero-mean, white, Gaussian noise. If the

noise is zero-mean but non-Gaussian, the Kalman Filter is the best *linear* estimator but not

the optimal estimator. Also, the standard Kalman Filter handles only one measurement vector at any time step. Although it is useful in many applications, these restrictions mean that the standard Kalman Filter presented above is not sufficient for my application. As expressed previously, because of both the uncertainty of each sonar image I attempt to match and the uncertainty on the map itself, simply taking the best match and calling that the solitary measurement may not provide the best results. Therefore, I opt for a set of the best matches as possible measurements and turn to data-association techniques to address the issues of uncertainty in measurement origin and of combining the information in the set of best matches to achieve the best estimated state.

3.2.2 Data Association

Data association (or data correlation) is typically employed in tracking algorithms to determine the best estimate of target location by appropriately associating each measurement with possible targets and evaluating the validity or usefulness of each measurement [Bar-Shalom and Fortmann; Fortmann et al.]. Many of the algorithms and processes of data association were originally developed for military applications such as tracking missiles, airplanes, and submarines. In these applications, measurement-origin uncertainty is usually due to random false alarms in detection, spurious measurements due to clutter, or interfering targets such as decoys. Data-association problems are usually classified by the type of association task:

Task 1. measurement-to-measurement association — track initiation

Task 2. measurement-to-track association — track maintenance or updating

Task 3. track-to-track association — track fusion in multi-sensor situations.

My application of navigation falls under Task 2, for which a variety of techniques can be used to determine the best update. Typically, these techniques can be divided into two categories: those that choose *one* of the measurements taken and those that combine measurements. The two most common single-measurement techniques are the strongest-neighbor standard filter (SNSF) and the nearest-neighbor standard filter (NNSF). The SNSF simply chooses the measurement giving the best match — the one with the lowest MAD (Match A of Figure 3.5). However, as indicated previously, this may not produce the
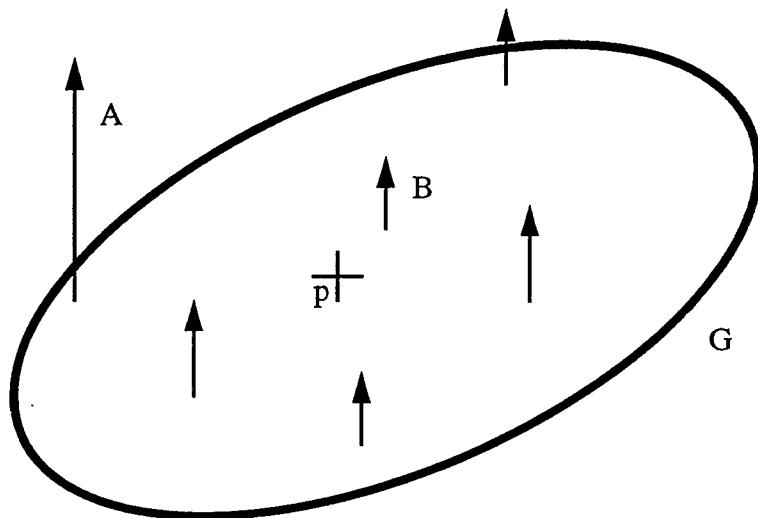
Figure 3.5. Single Measurement Association Techniques.

Arrow length indicates strength of six measurements in neighborhood of prediction *p* bounded by gate *G*. Strongest Neighbor Standard Filter selects *A*; Nearest Neighbor Standard Filter selects *B*.

best results. The NNSF uses the measurement closest to the prediction — the one with the smallest innovation (Match B of Figure 3.5). This measurement is chosen only for its proximity to the estimate, and the level of match "goodness" is ignored. This too may not produce the best results. Methods that combine measurements appear to be more

appropriate for this application, and my attention has focussed on stochastically combining the measurements in the set of best matches.

3.2.2.1 Probabilistic Data Association Filter

The Probabilistic Data Association Filter (PDAF) is a suboptimal Bayesian algorithm for situations in which there is only one target of interest, but several measurements of the target state are available at each time step [Bar-Shalom; Bar-Shalom and Fortmann]. A validation region or gate is defined around the predicted state, and a weighted sum of the measurements inside the gate is used to update the state. The PDAF operates using the Kalman Filter framework discussed above. The specific differences between these filters are explained below. The advantage of the probabilistic data association technique is that all measurements inside the validation region are considered with some probability of being correct. The disadvantage is that the resulting estimate is guaranteed to be "wrong" in the sense that the single correct measurement was not selected to stand alone. However, in my case, it is better to have an estimate with larger values of covariance than it is to choose the wrong measurement and have the AUV truly believe it is somewhere that it is not. What follows is a summary of the PDAF development as in [Bar-Shalom and Fortmann].

For each time step $k+1$, a validation region $V(\gamma)$, centered about the predicted measurement, is defined as

$$V(\gamma) \equiv \left\{ z; [z - \hat{z}(k+1|k)]'S^{-1}(k+1)[z - \hat{z}(k+1|k)] \leq \gamma \right\}. \qquad (3.12)$$

The validation region is an ellipsoid of probability concentration — the minimum volume that contains a given probability mass under the Gaussian assumption. The constant parameter γ can be obtained from tables of the Chi-square distribution if n_z, the dimension of the measurement, is known. The square root of γ, $g = \sqrt{\gamma}$, is often referred to as the "number of sigmas or standard deviation" of the gate. The set of valid measurements (i.e., those that fall inside the gate) is denoted by

$$Z(k + 1) = \{z_i(k + 1)\}_{i=1}^{m_{k+1}}, \tag{3.13}$$

where m_{k+1} is the number of valid measurements at time step $k+1$ and $z_i(k + 1)$ is the ith measurement. Similar to prior notation, Z^{k+1} represents all valid measurements over all time steps up to and including time step $k+1$, or

$$Z^{k+1} = [z(j)]_{j=1}^{k+1}. \tag{3.14}$$

The basic assumption of the PDAF is that the past is summarized by assuming that the state is normally distributed according to the latest estimate and covariance matrix, or

$$p[x(k + 1)|Z^k] = N[x(k + 1); \hat{x}(k + 1|k); P(k + 1|k)], \tag{3.15}$$

and the new estimate is found using the latest measurements only. The basic premise of the PDAF is that each validated measurement has a probability that it is the correct (target-originated) measurement and that there is also the probability that none of the valid measurements were target-originated. These events are defined as

$$\theta_i(k + 1) = z_i(k + 1) \text{ is the target-originated measurement } i = 1, 2, \dots m_{k+1},$$

$\theta_o(k + 1) =$ none of the measurements are target-originated,

and the probabilities associated with them are

$$\beta_i = P\left\{\theta_i \big| Z^{k+1}\right\} \text{ for } i = 0, 1, 2, ...m_{k+1}. \tag{3.16}$$

Since these events are mutually exclusive and exhaustive,

$$\sum_{i=0}^{m_{k+1}} \beta_i(k+1) = 1. \tag{3.17}$$

If $\hat{x}_i(k + 1 | k + 1)$ is the updated state estimate given that measurement $z_i(k + 1)$ is the correct one, and $v_i(k + 1)$ is the innovation for measurement $z_i(k + 1)$, then the updated state, given that measurement $z_i(k + 1)$ is correct, becomes

$$\hat{x}_i(k + 1 | k + 1) = \hat{x}(k + 1 | k) + W(k + 1)v_i(k + 1) \text{ for } i = 1, 2, ...m_{k+1}. \tag{3.18}$$

If none of the measurements is correct, then

$$\hat{x}_o(k + 1 | k + 1) = \hat{x}(k + 1 | k). \tag{3.19}$$

So, the conditional mean of the state estimate and the combined innovation are, respectively,

$$\hat{x}(k + 1 | k + 1) = \sum_{i=0}^{m_{k+1}} \hat{x}_i(k + 1 | k + 1)\beta_i(k + 1), \tag{3.20}$$

and

$$v(k+1) = \sum_{i=1}^{m_{k+1}} \beta_i(k+1)v_i(k+1). \qquad (3.21)$$

(Note that $v_o(k+1) = 0$.) Finally, the overall update equation is

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + W(k+1)v(k+1), \qquad (3.22)$$

where the gain $W(k+1)$ is found using the same method as in the Kalman Filter (Equation 3.11). Although Equation 3.22 and Equation 3.9 appear to be the same, the key difference is that here $v(k+1)$ represents the combined innovation of all measurements.

The main issues in developing the PDAF are to define the probabilities β_i and to use these weights appropriately when updating the state covariance matrix. The procedure that assigns the $\beta_i's$ is known as *probabilistic data association* (hence the filter name) and the equations are:

$$\beta_o(k+1) = \frac{b}{b + \sum_{j=1}^{m_{k+1}} e_j}, \qquad (3.23)$$

and

$$\beta_i(k+1) = \frac{e_i}{b + \sum_{j=1}^{m_{k+1}} e_j} \quad \text{for } i = 1, 2, \ldots m_{k+1}, \qquad (3.24)$$

where

$$e_i = \exp\left\{\frac{-1}{2}v_i'(k+1)S^{-1}(k+1)v_i(k+1)\right\}, \tag{3.25}$$

$$b = \left(\frac{2\pi}{\gamma}\right)^{n_z}\left(\frac{\lambda V(\gamma)}{c_{n_z}}\right)\frac{(1-P_D P_G)}{P_D}, \tag{3.26}$$

n_z is the dimension of the measurement, γ is the constant volumetric parameter (as in Equation 3.12), λ is the spatial density of false measurements, c_{n_z} is the volume of the n_z-dimensional unit hypersphere, P_D is the detection probability (probability that the correct measurement was detected at all), and

$$P_G = P\{z(k+1) \in V(\gamma)\} \tag{3.27}$$

is the gate probability (probability that the correct measurement fell inside the gate). The state prediction covariance becomes

$$P(k+1|k+1) = \beta_o(k+1)P(k+1|k) + \tag{3.28}$$
$$[1-\beta_o(k+1)]P^c(k+1|k+1) + P^t(k+1)$$

where

$$P^c(k+1|k+1) = [I - W(k+1)H(k+1)]P(k+1|k) \tag{3.29}$$

is the covariance of the state updated with the correct measurement (i.e., in the absence of measurement origin uncertainty), and

$$P^t(k+1) = W(k+1)\left[\sum_{i=1}^{m_{k+1}} \beta_i(k+1)v_i(k+1)v_i'(k+1)\right. \tag{3.30}$$

$$\left. - v(k+1)v'(k+1)\right]W'(k+1)$$

is the measurement-dependent term that accounts for the spread of validated measurements. Derivations and proofs of the above formulae can be found in [Bar-Shalom and Fortmann].

Before proceeding, I briefly summarize the status thus far. Because of the matching algorithm chosen and the uncertainty on the sonar image and the supplied map, it is deemed necessary to consider a set of good matches and not just the single best match. An algorithm is then needed to appropriately combine the information in each of these matches to determine a best estimate for the match location. The PDAF is a recursive sub-optimal Bayesian algorithm whose procedure is similar to that of a Kalman Filter. However, it differs in that the PDAF probabilistically combines multiple measurements at any time step and addresses the issue of uncertainty in measurement origin. The PDAF operates only on the current set of measurements and makes the assumption that the past can be approximately summarized by a state that is normally distributed according to the latest estimate and covariance matrix. Measurements in close proximity to the expected measurement value are weighted probabilistically as a function of the innovation, the detection probability, the gate probability, and the false-alarm rate. Thus, the new estimate is based on the information contained in multiple measurements. So, the basic advantage

of the PDAF is that all measurements in the vicinity of the expected value are considered with some probability of being correct.

3.2.2.2 Probabilistic Data Association with Amplitude Information

For my purposes, the main disadvantage of the PDAF is that once a measurement is validated the match 'goodness" of that measurement is no longer considered. To include this in my algorithm, I turn to the PDAF augmented with *amplitude information* (PDAFAI) [Lerro and Bar-Shalom] where I substitute the match "goodness" for raw signal amplitude. In fact, if handled appropriately, the "amplitude" can correspond to any feature or features that one may wish to use in determining measurement weights, such as level of backscatter or degree of bathymetric variability.

Adding the amplitude feature modifies the weight equations only because, although amplitude is technically another entry in the measurement vector, it is treated separately. Without proof or derivation, the new probabilistic weights are [Lerro and Bar-Shalom]

$$\beta_o(k+1) = \frac{b}{b + \sum_{j=1}^{m_{k+1}} e_j L_j} \qquad (3.31)$$

and

$$\beta_i(k+1) = \frac{e_i L_i}{b + \sum_{j=1}^{m_{k+1}} e_j L_j} \text{ for } i = 1, 2, \ldots m_{k+1}, \qquad (3.32)$$

where b, e_i, and m_{k+1} are as before. For applications where the actual signal amplitude is used, L_i is the amplitude likelihood ratio — the probability density function of the signal amplitude of a target above a given threshold with respect to the probability density function of noise above the same threshold. In my case, I use the match "goodness" as an amplitude feature, and the matching algorithm ranks the matches by the MAD.

Following the lead of [Lerro and Bar-Shalom], a Rayleigh probability density function is used to determine the appropriate correlation weights, L_i (see Figure 3.6). The
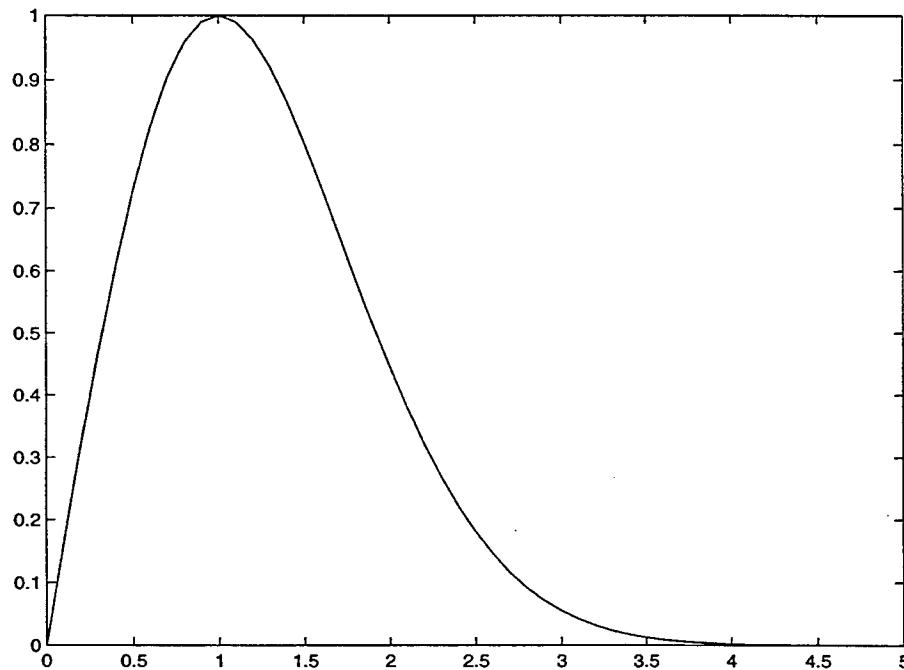
Figure 3.6. Rayleigh Distribution.
Normalized so that peak of unity is located at an abscissa of 1.

Rayleigh function

$$L_i = \exp\left(\frac{1}{2}\right)\frac{error_i}{\sigma_e}\exp\left(-\frac{error_i^2}{2\sigma_e^2}\right) \text{ for } i = 1, 2, \ldots m_{(k+1)} \tag{3.33}$$

is a one-sided distribution with an off-axis peak placed at the standard deviation of the error, σ_e. The coefficient *exp(1/2)* is incorporated so the peak of the distribution has a value of unity, and the parameter σ_e is recalculated at each time step as the one-sided standard deviation of the errors of all validated measurements. With this formulation, the weights L_i are relative weights (relative to one another) and not absolute weights (relative to a perfect match). Note that the sum of all β_i including β_o remains one but that the sum of all L_i is not one. Also, the relative emphasis between the measurements and the prediction can be manipulated by changing the ratio of the terms in the denominator of the β_i equations,

$$\frac{\displaystyle\sum_{j=1}^{m_{k+1}} e_j L_j}{b}.$$

A high value of this ratio puts emphasis on the measurements, and a low value puts emphasis on the prediction. Regardless, manipulation of this ratio will not lead to significantly different results, and such effort in this area was not determined worthwhile at this time.

3.2.2.3 Parameters

In addition to the transition matrices, control inputs, covariance matrices, and initial conditions, required input parameters for the updating algorithm are g, λ, and P_D. Parameters such as P_G, n_z, and c_{n_z} are determined by the dimensionality of the application or by a combination of other parameters. For example, the gate probability P_G is determined by n_z, g, and the Chi-square distribution (see Table 3.1). The gate size in sigmas, g, is arbitrarily selected but usually falls between two and five, depending on the application and how conservative the filter should be. I choose $g=4$ so there is a very

small probability of rejecting measurements because of innovation magnitude. Since the methods of this application are somewhat different from a traditional radar/tracking implementation, the remaining two parameters, λ and P_D, warrant a more thorough explanation.

Table 3.1 Gate Probabilities

	g=1	g=2	g=3	g=4	g=5
$n_z = 1$	.683	.954	.997	.99994	1.0
$n_z = 2$	.393	.865	.989	.9997	1.0
$n_z = 3$	.199	.739	.971	.9989	.99998

In a traditional tracking problem, P_D is the probability that the correct measurement is detected at all, regardless of whether or not it falls within the gate [Bar-Shalom and Fortmann; Van Trees]. The detection probability is incorporated because there is some chance that none of the measurements taken is actually target-originated. The value used for P_D is usually determined by the quality of the detection device and the environment of operation. In my application the target of interest is the match location that gives the right answer to the navigation problem. As stated in Section 3.1.1, when using a coarse-to-fine matching algorithm, it is possible that the right measurement is missed due to the low-pass filtering and downsampling. It is also possible that the right match does not satisfy the threshold criteria due to the noise on the sensor and the inaccuracies of the map itself. Additionally, there is a probability that the right match lies outside the established search area and is then never detected. As [Bergem] says, analytically determining a value for this parameter can be a difficult task.

The remaining parameter, λ, represents the spatial density of false measurements. A false measurement or false alarm is defined as a completely erroneous or spurious measurement, i.e., a measurement not associated with the target of interest. Target-originated measurements with considerable noise do not qualify as false measurements. In a radar tracking application, false alarms usually result from other targets (clutter) in the environment or multipath reflections [Bar-Shalom and Fortmann; Van Trees]. In the terrain-relative navigation application, false measurements can arise when the scanned image is matched to the map in an inappropriate way. This occurs when there is something incorrect about the image or the map, such as a bias or the absence (presence) of features that do (do not) belong, or when several places on the map are similar because of the map resolution. If a section of the seafloor image is incorrect, say, as a result of multipath or acoustic interference from other sources or sensors, then *all* associated matches can be incorrect. In this case, the best action to take might be to ignore all matches and proceed with the next scan. However, without complex analysis beyond the current scope, we do not know this is a bad scan and can only proceed as best as data association allows. If instead a portion of the map is incorrect, say, a seamount in the real world does not appear in the map, only matches that involve this portion will produce false measurements. So, in this sense, λ is influenced by the map accuracy and image accuracy. A good map (image) produces few bad measurements, so λ should be small, but a poor map (image) produces many false matches, so then λ should be large. If the map and image are supplied not only with bathymetry values but also with measures of uncertainty on those values, then it is with the parameter λ that we can include this uncertainty in the updating algorithm.

3.3 Summary

In this chapter I present the two underlying principal algorithms of terrain-relative navigation. The matching algorithm uses the mean absolute value difference to determine match quality and searches the neighborhood using the coarse-to-fine procedure with a modified beam search. The PDAFAI is used to provide navigation updates based on both a predicted position and the information in validated measurements supplied by the matching algorithm. Each validated measurement is weighted probabilistically as a function of both its distance from the prediction and the match quality. To achieve navigation, these two algorithms are merged. The next chapter addresses the interrelationships between the critical parameters.

# Chapter 4

Merging the Algorithms

All major pieces necessary to achieve localization are presented in the previous chapter. The research task remains to integrate these algorithms into an appropriate process. For this research the sonar image is a one-dimensional profile represented as a list of depth values, equally spaced, with the same spacing as the pixels of the bathymetric map. The state vector is

$$\hat{x}(k+1|k) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, \qquad (4.1)$$

where x and y indicate the column and row, respectively, of the matrix, which represents the digital map, and θ is the orientation of the profile with respect to the x axis, i.e., a matrix row. For simplicity in all simulations, the state is measured directly, so the transformation matrix $H(k+1)$ (see Section 3.2.1) is the identity matrix, and the measurement vector is

$$z(k+1) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}. \qquad (4.2)$$

The overall navigation process follows the Kalman Filter framework shown in Figure 4.1. First, the state and state covariance are predicted by an internal navigation system,

| Prediction | 1 | 1. Predict state, state covariance, and measurement
    Calculate innovation covariance and Kalman Gain |

```
┌─────────────────────┐     1. Predict state, state covariance, and
│  Prediction      1  │     measurement
└─────────────────────┘          Calculate innovation covariance and
          │                      Kalman Gain
          ▼
┌─────────────────────┐     2. Determine search parameters as a
│ Search Parameters 2 │     function of innovation covariance
└─────────────────────┘
          │
sonar profile
          │
       ──────►
       ──────►┌──────────────────┐  3. Employ coarse-to-fine matching
              │  Matching     3  │  algorithm with beam search
       ──────►└──────────────────┘
map       │
          ▼
┌─────────────────────┐     4. Validate measurements inside the gate
│  Validation      4  │
└─────────────────────┘
          │
          ▼
┌─────────────────────┐     5. Update state estimate and state
│  Update          5  │     covariance using data association
└─────────────────────┘
```

Figure 4.1. Kalman Filter Framework.

See also Figure 3.2.

such as dead reckoning or inertial navigation (see Section 2.1). From this information, the

search area of the map is determined, then the matching algorithm is employed to produce

a list of measurements by matching the profile to the map. Finally, the state estimate and

state covariance are updated using the data-association techniques presented in Chapter 3.

The current chapter shows how the matching and updating algorithms are interrelated

through the search range, the probability of detection P_D, the spatial density of false

alarms λ, and the measurement noise covariance matrix $R(k+1)$.

4.1 Search Area

The search area at time step $k+1$ is found from the innovation covariance $S(k+1)$ by creating an ellipse around the estimated position. The size and orientation of the ellipse are determined by the covariances in and the covariance between the x and y directions. The ellipse is defined at g standard deviations, where $g=4$, (see Section 3.2). For simplicity in implementation, the search area is defined as the smallest rectangle aligned with the map grid that encompasses the ellipse (see Figure 4.2). For each pixel in the search area,
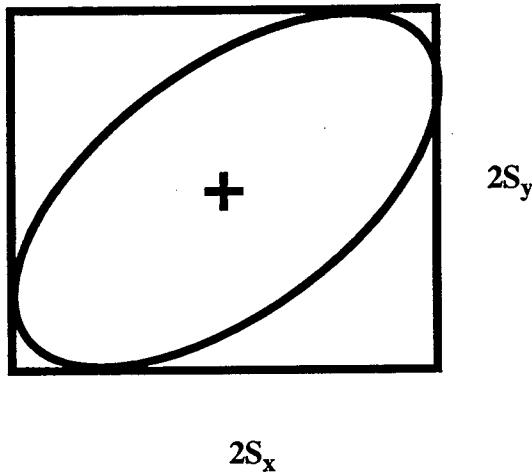
Figure 4.2. Ellipse and Search-Area Rectangle.

the full range of search angles, determined by $S_{\theta\theta}(k+1)$ and g, is tested. Also, any pixel with as little as a single corner located inside this rectangle is included in the search area, even though the actual pixel location is defined as the pixel center, which may be outside the rectangle. This means that there are matching attempts made in the search area that are not located inside the validation gate, and good matches found in this region are rejected later by the PDAFAI. Additionally, since the coarse-to-fine algorithm has flexible boundaries, some pixels searched in finer levels may also lie completely outside the

validation gate (see Section 3.1.1). For example, if at a coarse level one of the top w matches corresponds to a pixel at the edge of the gate, then, when moving to the next finest level, that node is still expanded using all nine branches, even though some of these branches are outside the originally specified validation gate. As above, good matches found outside the gate are rejected later by the PDAFAI.

The size of the established initial search area determines the amount of downsampling used for that time step. If the search area is defined as $x \pm S_x$ and $y \pm S_y$, where x and y are the estimates, and S_x and S_y are half the length and width of the encompassing rectangle (see Figure 4.2), then the numerical value of interest is $S_x S_y$, which represents 1/4 of the pixels in the search area. By evaluating the simulation run time for a variety of cases, the guidelines of Table 4.1 were empirically deduced so the size of the search area determines the amount of downsampling.

Table 4.1 Search Area vs. Downsampling

$g^2 S_x S_y$	<12	12< <=48	48< <=192	192<
Downsampling	1	2	4	8

4.2 Detection Probability and Spatial Density of False Alarms

After searching and determining the set of measurements used in updating, the modified PDAFAI is used, and the values for the probability of detection, P_D, and the

spatial density of false alarms, λ, must be determined (see Section 3.2.2.3). First, we state

that λ is related to the probability of false alarm by the equation

$$\lambda = \frac{P_F}{V_c},$$ (4.3)

where P_F is the probability of false alarm for a resolution cell of the processor, and V_c

is the resolution-cell volume. If the resolution cell is defined as one pixel in length by one

pixel in width by one increment in orientation, then λ can be determined if P_F is known.

Now, the problem at hand is determining the detection and false-alarm probabilities.

In binary hypothesis testing for signal-processing applications, four probabilities arise

[Bar-Shalom and Fortmann; Van Trees]:

P_D — Detection Probability — we say it is there when it is,

P_F — False Alarm Probability — we say it is there when it is not,

P_M — Miss Probability — we say it is not there when it is,

Unnamed Probability — we say it is not there when it is not.

Historically, the fourth probability has been left unnamed, but it is equal to $1 - P_F$. Also,

$P_M = 1 - P_D$. These probabilities are most often determined by defining the

probability density functions of the noise involved in the particular application, then

integrating to the left or the right of a threshold determined by the signal-to-noise ratio.

My application differs from the typical situation in which this is applied, so I adapt the

standard approach to be more suitable.

In my application of matching profile to map, each profile point has an uncertainty of σ_p, and each map point has an uncertainty of σ_m. The absolute value error between profile and map depth values is then used to determine whether that particular location gives a sufficiently good match. Consider first the profile as a single point, so the error is found by determining the absolute difference between this point and the corresponding point on the map. If we sit on the profile point and absorb all the uncertainty of the map [Smith et al.], the result can be depicted as in Figure 4.3, where P represents the profile value, M the map value, and σ^2 is the total uncertainty $\sigma^2 = \sigma_p^2 + \sigma_m^2$. Equivalently, we can shift the axis so the Gaussian uncertainty is centered about the origin, and the difference between P and M is represented by E. Now, if the profile is a series of points, all with uncertainty σ_p, and the map is also a group of points, all with uncertainty σ_m, then the depiction of Figure 4.3 is still valid if E now represents the average absolute value difference across all the points. This value is the mean absolute difference (MAD) used as the goodness criteria in the matching algorithm (see Section 3.1).

Next, let us say that the best match has a MAD value of E, and the maximum validated MAD, as defined by the percentage criteria, is value F, and we redraw the figure as in Figure 4.4. Now, no matches have MAD smaller than E, all matches with MAD values between E and F are validated, and all other matches have MAD greater than F. More specifically, for the right answer to be validated, it must have a MAD value between E and F, and for it to be missed or rejected, it must have a MAD value greater than F. Using Figure 4.4, the probability of a miss is found by integrating the area under the uncertainty

P or 0   M or E

depth of or depth difference between profile and map values
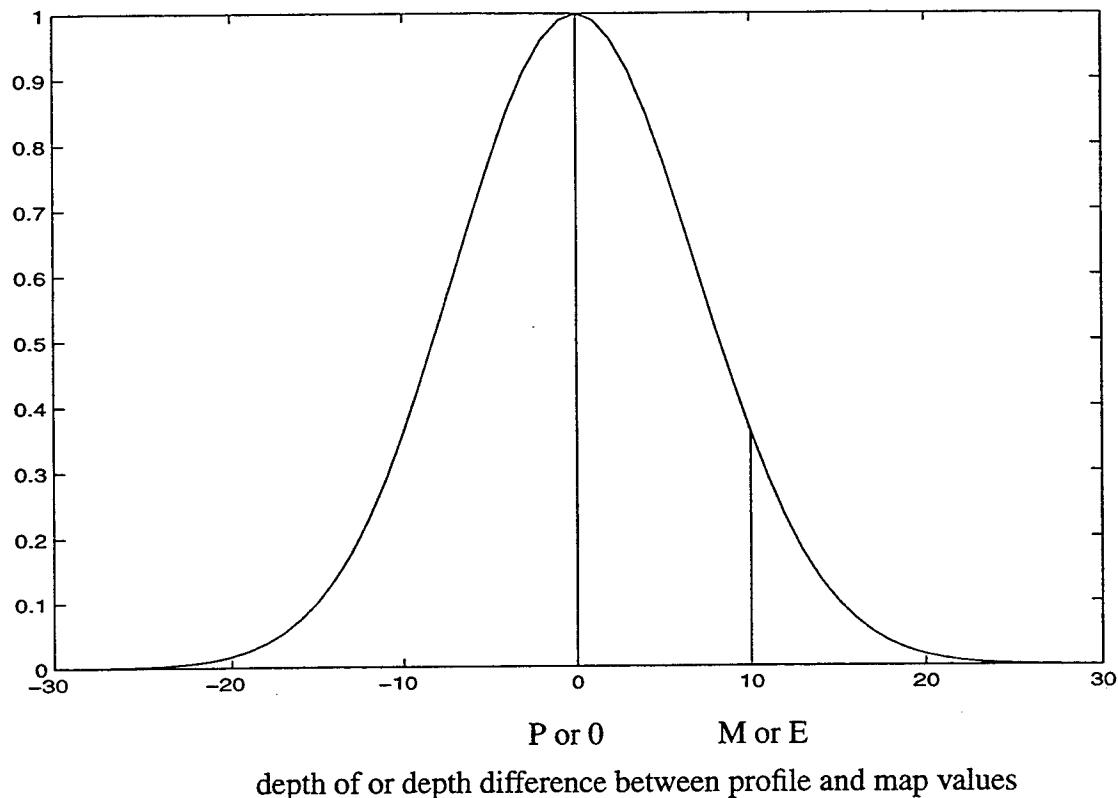
Figure 4.3. Total Gaussian Uncertainty Placed on Profile.

With deviation σ – about profile depth value P with corresponding map depth value M, or about zero with average absolute value difference E between all profile and map points.

curve from F to ∞. Actually, because all MAD values are positive, the true miss probability is

$$P_M = 2\int_{F}^{\infty} G(e)de, \tag{4.4}$$

where $G(e)$ is the Gaussian uncertainty curve as a function of the MAD e. The detection probability is
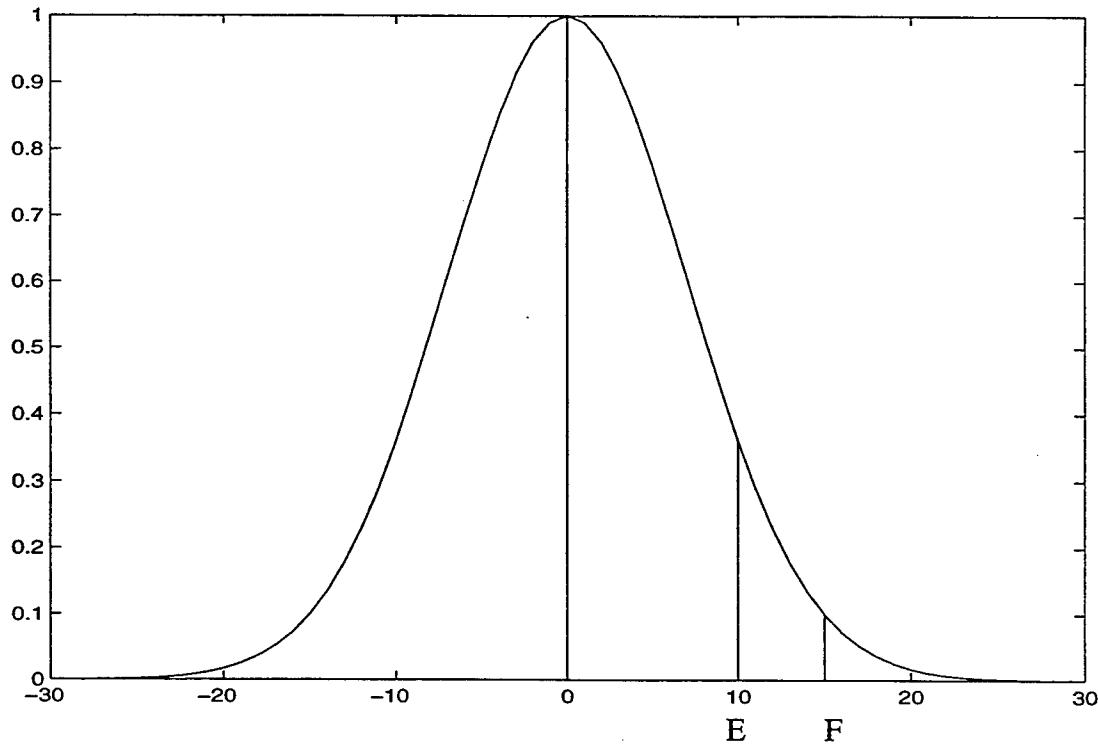
depth difference between profile and map values

Figure 4.4. Total Gaussian Uncertainty Placed on Profile with Acceptance Thresholds. With deviation σ - best match has error E, validated match with maximum error has error F. No matches with error between 0 and E. All validated matches have error between E and F. Missed and rejected matches have error greater than F.

$$P_D = 1 - 2\int\limits_{F}^{\infty} G(e)de.$$ (4.5)

This value must then be multiplied by the gate probability, P_G (Equation 3.27), because

we approximate the search region as the area within the gate. Finally, the equation we use

is

$$P_D = P_G\left(1 - 2\int\limits_{F}^{\infty} G(e)de\right).$$ (4.6)

70

In the standard application [Van Trees], as the acceptance threshold is raised higher, more of the signal is rejected because the acceptance criterion is more stringent. Thus, the detection probability is reduced. Here, the raising of the threshold corresponds to the shifting of E, and consequently F, towards the origin, meaning that the MAD must be smaller to pass the acceptance criterion. This results in a larger calculated value for P_M, and thus a smaller P_D, as it should be. For a given value of E, varying the acceptance criterion, or percentage value (see Section 3.1.1), changes the location of F. For small values of the percentage threshold, F is closer to E, and the integration of Equation 4.4 begins closer to the origin. Thus, P_M becomes larger and P_D smaller. Conversely, for large values of this threshold, P_M decreases and P_D becomes large.

Next, since only the matches with MADs between E and F have been validated, all false alarms must be part of this set and also have MADs between E and F. Further, since there is only *one* right answer (see Section 3.2.2.1), all matches between E and F, except potentially one, are false alarms. We therefore approximate the false-alarm probability as twice the area under the uncertainty curve between E and F, knowing that this can potentially be a slight over-estimate:

$$P_F = 2\int_E^F G(e)de. \tag{4.7}$$

As with the detection probability in the standard application [Van Trees], raising the acceptance threshold reduces the false-alarm probability, as more of the signal is filtered out. For my application, shifting E and F toward the origin means that P_F is found by

integrating over a part of the curve that has greater magnitude. This seemingly gives a higher value; however, because F is found by the *percentage* criterion (see Section 3.1.1), the separation between E and F becomes smaller as E moves toward zero. So, although the magnitude of the probability curve is higher, the distance over which it is integrated is smaller. Additionally, as with the detection probability, reducing the percentage threshold value reduces the value of P_F, because for a given value of E the integration is over a smaller range. Similarly, for larger percentage values, the integration produces a larger value for P_F.

The specific nature of P_F with respect to the values of E and F is also dependent on the magnitude of the variance σ^2. The false-alarm probability does not follow the same trend as the curve for the standard application when σ is small and E is large, or, in other words, when the noise should be small but the MAD values are actually quite large. When this occurs it is an indication that something is very wrong, such as a bad bias in the profile or map, or the matching process is being applied to the wrong area of the map, i.e., the vehicle is not where it thinks it is.

4.3 Measurement Noise Covariance Matrix

The 3x3 measurement noise covariance matrix $R(k+1)$ represents the zero-mean white Gaussian measurement noise on the combined match location (x, y, θ). This matrix is used to determine the Kalman Gain $W(k+1)$ of the PDAFAI (Equations 3.8, 3.11, and 3.22), which defines the relative weight between the prediction and the combined information

from the validated measurements. The uncertainty defined by $R(k+1)$ can be visualized as an ellipsoid not necessarily aligned with the axes of the coordinate frame. The off-diagonal terms indicate the cross-covariance between the three types of noise, and they determine the 3D orientation of the ellipsoid. Only if all off-diagonals are zero will the ellipsoid and frame axes be aligned.

In most applications, the entries in the measurement noise covariance matrix are found directly from the sensor taking the measurement. However, in this application the measurement is found by the matching process. For this reason, we find the entries of the $R(k+1)$ matrix empirically from the process and data themselves. We must, however, also make one assumption: that all valid measurements at a single time step have the same uncertainty and that uncertainty is the same as the uncertainty on a measurement located at the prediction. That multiple measurements are used in determining the combined measurement is not relevant for defining $R(k+1)$; that information is accounted for elsewhere in the updating procedure by using the measurement weights β_i (Equation 3.32) and the state covariance spreading term $P^t(k+1)$ (Equation 3.30). This assumption allows us to use the PDAFAI formulation described previously, because in that procedure it is assumed that all measurements have the same uncertainty.

Now, the task at hand is to determine the measurement uncertainty at the location of the prediction. To approximate this empirically we use the following procedure. First, we generate an exact profile at the predicted location by sampling the known map. Second, using this profile we apply the matching algorithm to the neighborhood around the prediction and evaluate the matching strength. With this information and the noise variance on the profile, $R(k+1)$ can be determined using an adaptation of the Cramer-Rao

Lower Bound, as described below. Doing this, the measurement noise covariance matrix becomes, as it should be, a function of the terrain, the matching process, and the profile uncertainty.

4.3.1 The Cramer-Rao Lower Bound

The Cramer-Rao Lower Bound (CRLB), the most extensively used bound in the signal-processing literature, is often employed to obtain the resolution of measurements [Willsky et al.]. It is defined as the lower bound on the variance about a parameter A of any unbiased estimator. The bound is expressed as the inverse of the sensitivity, which is defined as the curvature of a selected function $f(A)$ about the parameter value of interest A_o. Pictorially, the CRLB corresponds to fitting a quadratic at the peak of the curve A_o (see Figure 4.5). Visually, one can see that if the peak is sharp a slight variation in the
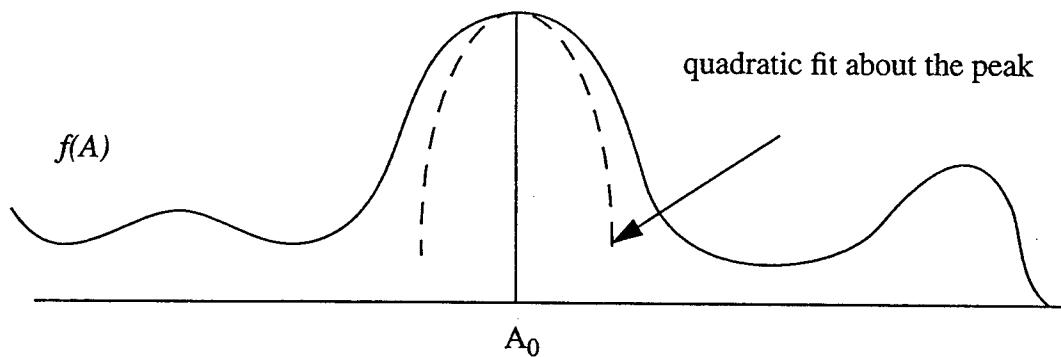
Figure 4.5. Cramer-Rao Lower Bound.
Determined by fitting a quadratic to the peak of $f(A)$ at A_0.

parameter value can produce a large difference in the function value. This means that the system is very sensitive, so the CRLB must be small. Likewise, if the peak is flat, a large variation in the parameter value is required for a large difference in the function value. Here, the system is not sensitive to the parameter, and thus the CRLB should be large.

There are two things that should be noted now. First, the function $f(A)$ is usually determined by an auto- or cross-correlation function or the correlation coefficient. Second, the CRLB represents the optimistic minimum resolution or uncertainty, but in some cases this bound is far from attainable. This is true because the curvature at the peak determines the bound independently of the overall shape of the function.

The CRLB is inappropriate for my application for two reasons. First, when calculating parameters empirically, it is important to use the same function as in the overall process that uses the parameter. For me, this translates to using the same similarity parameter function as in the matching algorithm (see Section 3.1), namely the mean absolute difference instead of the correlation function. Second, for my application I require a realistic estimate for the measurement uncertainty, not a lower bound, and most definitely not an unattainable lower bound.

4.3.2 An Adaptation

Values for the measurement noise covariance matrix can be determined using the concepts of the CRLB if a few adjustments are made. As stated above, the MAD should be used instead of the correlation function. The MAD is actually a dissimilarity parameter [Romesburg], which ranges from zero for perfect similarity to infinity for large dissimilarity. When the profile is matched across a map row or column, this parameter gives a valley instead of a peak at the position of perfect match, and thus the fitted quadratic must open upward instead of downward (see Figure 4.6). Here, we outline the procedure for defining the quadratic in one dimension and subsequently expand the concept to address multidimensionality.
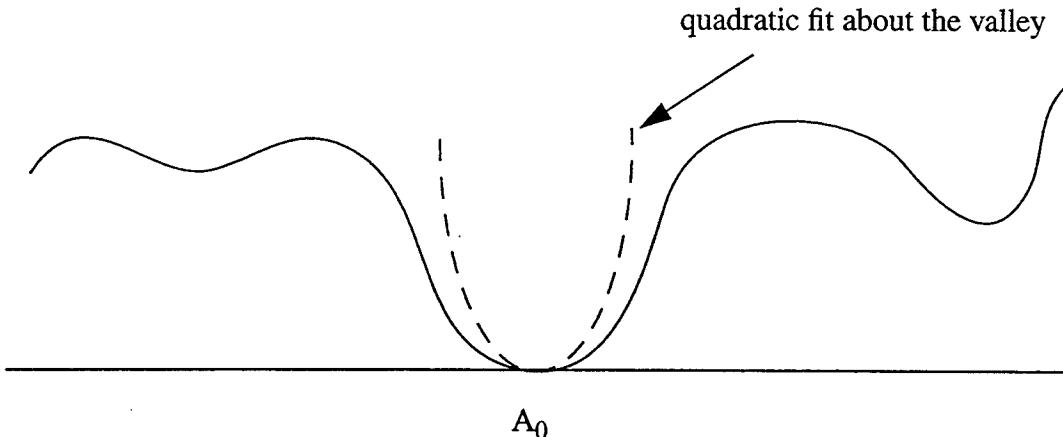
Figure 4.6. Quadratic Fit at A_0 opening upward.

For a chosen location x_0 on the map, a profile centered about x_0 is selected. The profile matches the map exactly at this location since no noise is added. The profile is then matched across the whole map row (x-direction) to give the dissimilarity parameter function (DPF), as in Figure 4.7. Since in reality there will be Gaussian noise on each point, a threshold d that represents this value is also drawn in the figure. The two points on the function nearest to the valley minimum and on opposite sides of x_0 which intersect this threshold are noted as x_1 and x_2. Next, the maximum of $|x_0 - x_1|$ and $|x_0 - x_2|$ is chosen. Finally, a parabola that intersects this point and has its valley minimum at x_0 is obtained. The inverse of the curvature of this parabola at x_0 gives the resolution or variance in the x direction, σ_x^2.

Selecting an intersection point between the threshold and the DPF assures that the calculated resolution is not an unattainable lower bound. In fact, choosing the further of the two points will not create any type of lower bound but may in fact give a value closer
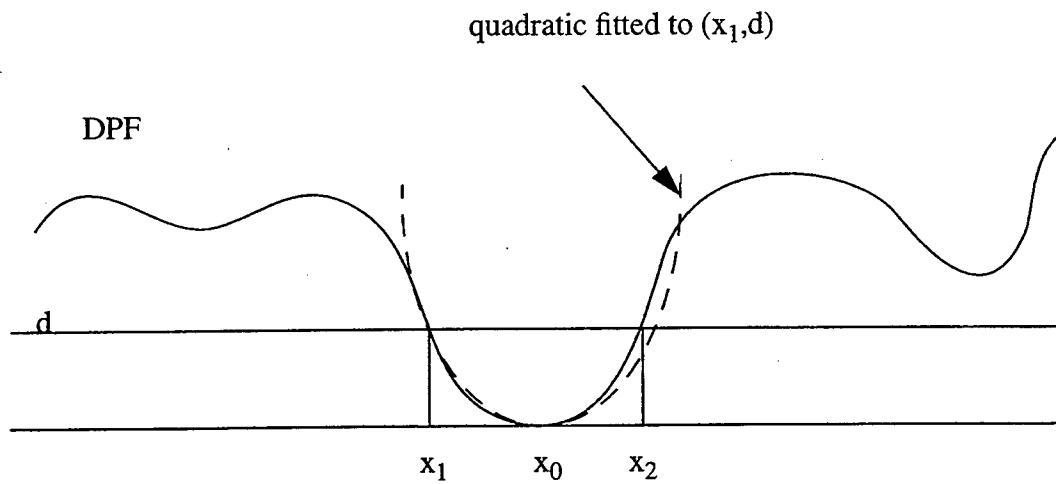
Figure 4.7. Quadratic Fit of Dissimilarity Parameter Function (DPF) with $|x_1 - x_0| > |x_2 - x_0|$. to an upper bound. We cannot, however, claim that this value is a true or strict upper bound.

To obtain all entries of the 3x3 measurement noise covariance matrix, the above technique is expanded for three two-dimensional cases and, instead of a parabola, an elliptic paraboloid is fitted. For example, consider the two-dimensional case for the x and y directions. The selected profile is then matched across both directions, and a dissimilarity parameter surface (DPS) is defined, still with valley minimum at the chosen location of (x_0, y_0). The threshold, d, is then a plane that intersects this surface in a contour instead of points (see Figure 4.8). From the contour, the points (x_1, y_0), (x_2, y_0), (x_0, y_1), and (x_0, y_2) are easily found, and the maximum distances in the x and y directions are taken as before. Additionally, the point on the contour with maximum distance from (x_0, y_0) is labeled (x_m, y_m). These three points are then substituted into the equation for an elliptic paraboloid as follows.
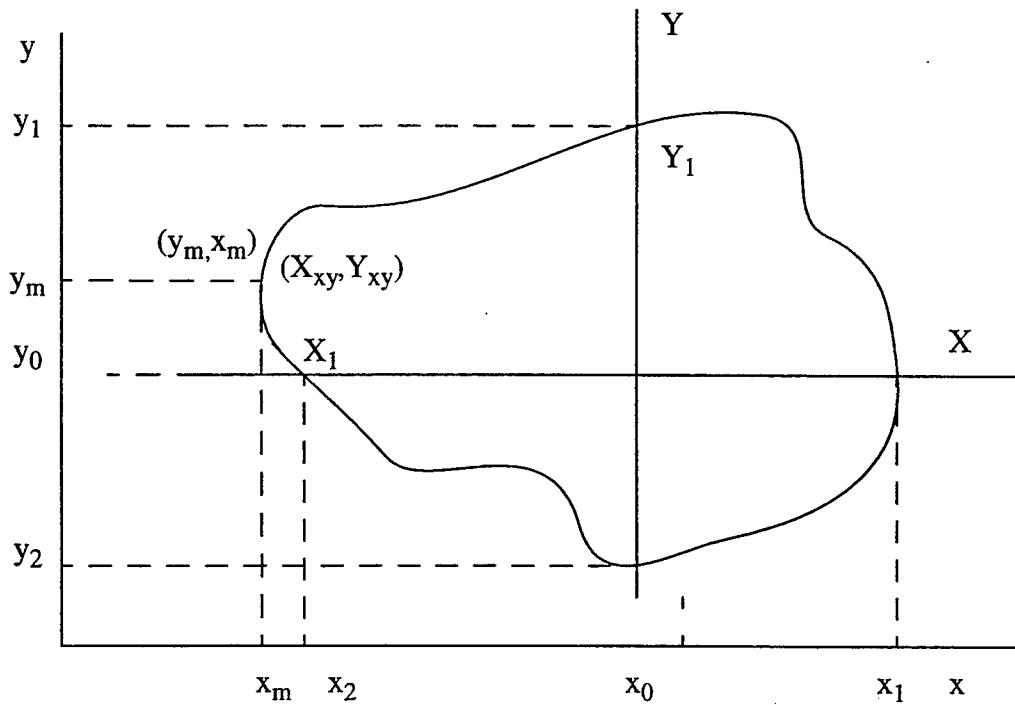
Figure 4.8. Intersection Contour at Threshold Showing Key Points.

First, we establish a coordinate frame with origin at $(x_0, y_0, 0)$ and label the axes X, Y,

and Z. Next, the three points selected are defined as $(X_1, 0, d)$, $(0, Y_1, d)$, and (X_{xy}, Y_{xy}, d), where X_1 is the maximum of $|x_0 - x_1|$ and $|x_0 - x_2|$, Y_1 is the maximum of $|y_0 - y_1|$

and $|y_0 - y_2|$, and (X_{xy}, Y_{xy}, d) is $(x_m - x_0, y_m - y_0, d)$. The elliptic paraboloid is defined

by

$$AX^2 + BXY + CY^2 + Z = 0, \tag{4.8}$$

where A, B, and C are constants. Substituting in the three points above determines the

values of these constants, and the equation becomes

78

$$\frac{d}{X_1^2}X^2 - \frac{d}{X_{xy}Y_{xy}}\left(\frac{X_{xy}^2}{X_1^2} + \frac{Y_{xy}^2}{Y_1^2} - 1\right)XY + \frac{d}{Y_1^2}Y^2 = Z. \qquad (4.9)$$

As an example, Figure 4.9 shows the special case where the elliptic paraboloid is aligned
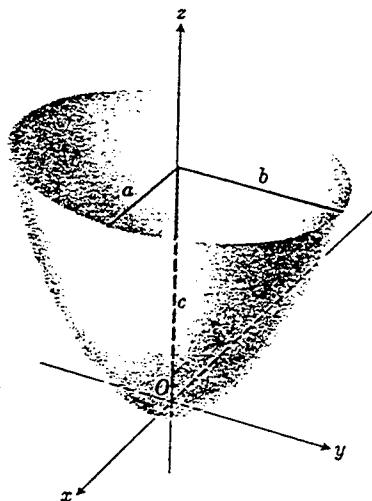
Figure 4.9. Elliptic Paraboloid Aligned with Axes.
$a=X_1$, $b=Y_1$, $c=d$, from [Thomas and Finney].

with the axes so there is no XY cross term.

Finally, the curvatures (second derivatives) are taken about $(0, 0, 0)$:

$$\frac{\partial^2 Z}{\partial X^2} = \frac{2d}{X_1^2}, \qquad (4.10)$$

$$\frac{\partial^2 Z}{\partial Y^2} = \frac{2d}{Y_1^2},$$

(4.11)

and

$$\frac{\partial^2 Z}{\partial X \partial Y} = \frac{-d}{X_{xy} Y_{xy}} \left(\frac{X_{xy}^2}{X_1^2} + \frac{Y_{xy}^2}{Y_1^2} - 1 \right),$$

(4.12)

and the variances σ_x^2, σ_y^2, and σ_{xy}^2 are the inverses of these three equations, respectively.

Fundamentally, this procedure fits an ellipse to the contour defined by d and, in turn, fits the paraboloid to this ellipse.

For the orientation terms, an elliptic paraboloid is created for each of the two-dimensional cases $X\Theta$ and $Y\Theta$. For the $X\Theta$ case, the equation is

$$\frac{d}{X_1^2} X^2 - \frac{d}{X_{x\theta} \Theta_{x\theta}} \left(\frac{X_{x\theta}^2}{X_1^2} + \frac{\Theta_{x\theta}^2}{\Theta_1^2} - 1 \right) X\Theta + \frac{d}{\Theta_1^2} \Theta^2 = Z,$$

(4.13)

and the parameters X_1, Θ_1, $X_{x\theta}$, and $\Theta_{x\theta}$ are defined by matching across the x direction

and rotating in Θ. Here, X_1 is the same as in the XY case, but $X_{x\theta}$ is not the same as X_{xy},

because it is determined by the $X\Theta$ contour at d instead of the XY contour.

The last equation, for the $Y\Theta$ case, is

$$\frac{d}{Y_1^2}Y^2 - \frac{d}{Y_{y\theta}\Theta_{y\theta}}\left(\frac{Y_{y\theta}^2}{Y_1^2} + \frac{\Theta_{y\theta}^2}{\Theta_1^2} - 1\right)Y\Theta + \frac{d}{Y_1^2}Y^2 = Z, \qquad (4.14)$$

and Y_1 and Θ_1 are as before, but $Y_{y\theta}$ and $\Theta_{y\theta}$ are determined by the $Y\Theta$ contour at d.

Finally, the remaining three variances σ_θ^2, $\sigma_{x\theta}^2$, and $\sigma_{y\theta}^2$ are, respectively, the inverses

of

$$\frac{\partial^2 Z}{\partial \Theta^2} = \frac{2d}{\Theta_1^2}, \qquad (4.15)$$

$$\frac{\partial^2 Z}{\partial X \partial \Theta} = \frac{-d}{X_{x\theta}\Theta_{x\theta}}\left(\frac{X_{x\theta}^2}{X_1^2} + \frac{\Theta_{x\theta}^2}{\Theta_1^2} - 1\right), \qquad (4.16)$$

and

$$\frac{\partial^2 Z}{\partial Y \partial \Theta} = \frac{-d}{Y_{y\theta}\Theta_{y\theta}}\left(\frac{Y^2_{y\theta}}{Y^2_1} + \frac{\Theta^2_{y\theta}}{\Theta^2_1} - 1\right). \qquad (4.17)$$

Note that in all cases the curvatures in the X, Y, and Θ directions are positive, giving

positive entries along the diagonals of $R(k+1)$, but that the "cross-curvatures", and thus the

off-diagonals, may be either positive, negative, or zero. Then, the 3x3 measurement

covariance matrix is defined as

$$R(k+1) = \begin{bmatrix} \sigma^2_{xx} & \sigma^2_{xy} & \sigma^2_{x\theta} \\ \sigma^2_{xy} & \sigma^2_{yy} & \sigma^2_{y\theta} \\ \sigma^2_{x\theta} & \sigma^2_{y\theta} & \sigma^2_{\theta\theta} \end{bmatrix}. \qquad (4.18)$$

The final step in the procedure is to check if $R(k+1)$ meets the criterion for a realistic

covariance matrix. Cross-covariances of a true covariance matrix are limited by the

correlation coefficient ρ_{ij} [Papoulis], which is defined as

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i\sigma_j} \qquad (4.19)$$

for the variables i and j, and this similarity parameter [Romesburg] has range

$$|\rho_{ij}| \leq 1. \qquad\qquad (4.20)$$

Thus, given σ_i and σ_j, minimum and maximum bounds are set for σ_{ij}. Consequently, if

the above procedure calculates a value for a cross-covariance outside the acceptable range,

the R(k+1) matrix is not a valid covariance matrix. When this occurs, the cross-covariance

is set to the appropriate minimum or maximum value so an acceptable matrix is used in

the updating algorithm, (see Section 3.2).

4.3.3 Examples

Next, this technique is demonstrated using examples from real bathymetry. Figure

4.10 shows a contour plot of deep-ocean bathymetry approximately located at $26.35°N$
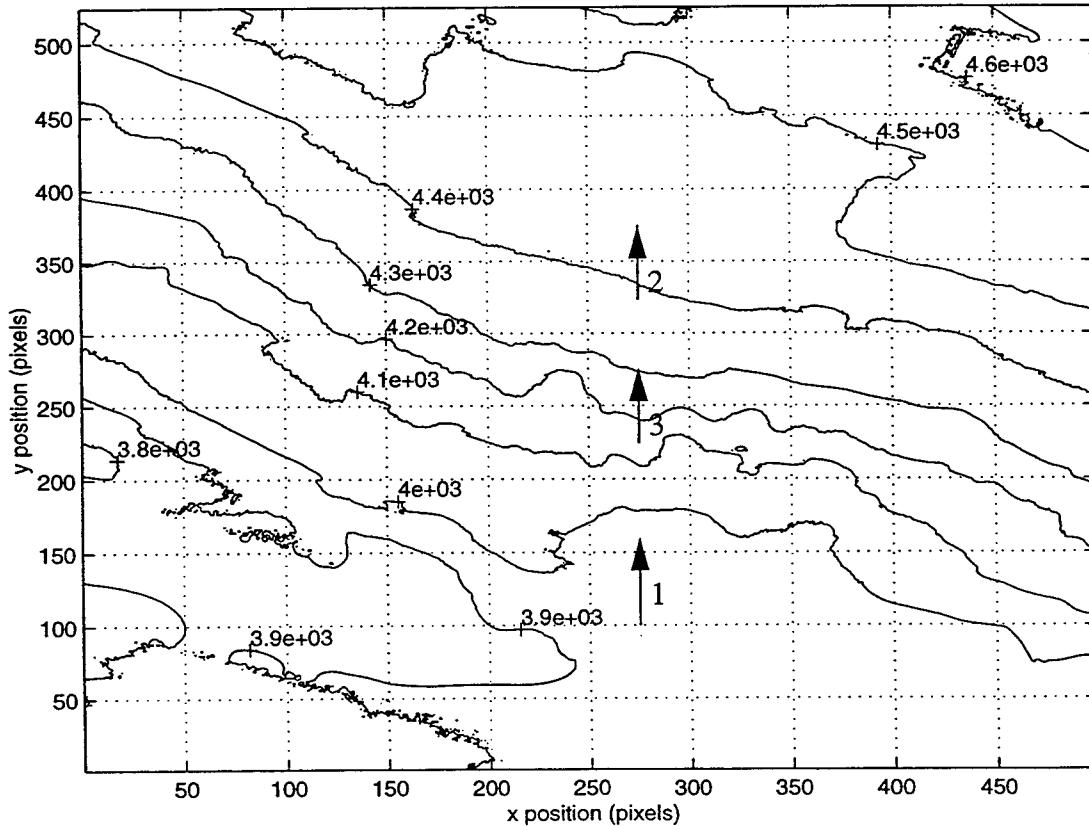
Figure 4.10. Contour Plot of Deep Ocean Bathymetry with 3 Example Profiles.

83

$46.05° W$ on the western flank of the Mid-Atlantic Ridge between the Kane and Atlantis Transforms. The map was created from data taken by a 120-kHz sidescan sonar. The region shown is a 2495 m x 2620 m section of a steep ridge bounded on one side with a sedimentary plateau and on the other by an area of gradual slope. The pixel spacing is 5 m in both directions, and the minimum and maximum depths are 3769 m and 4631 m, respectively. Three profiles from this map, also drawn on the figure, were chosen to demonstrate how the terrain affects values of the measurement noise covariance matrix. The profiles are centered about (275, 125), (275, 350), and (275, 250), and all have an orientation value of $90°$, which means they correspond to a subset of a column.

Figure 4.11 shows an enlargement of the map for the neighborhood around the first profile, which is given in Figure 4.12. This region is the flattest area of the overall map. Figures 4.13, 4.14, and 4.15 show respectively the contours at $d=5$ for XY, $X\Theta$, and $Y\Theta$. The XY contour roughly follows the terrain in this region as it should, and the associated three $R(k+1)$ values are $R_{xx} = 80$, $R_{yy} = 108$, and $R_{xy} = 146$. Here, R_{xy} exceeds the acceptable limit defined by the correlation coefficient, so R_{xy} is set to the maximum acceptable value, or $R_{xy} = 93$. The $X\Theta$ and $Y\Theta$ contours indicate that the matching process is very insensitive to the orientation at this location. In fact, these contours do not close because it is so insensitive that even a $60°$ difference in orientation is not sufficient to produce a large enough value of the DPS.

For these cases, it is not possible to obtain a true value for θ_1 and θ_2, so instead the maximum value of $60°$ is chosen as Θ_1, and $R_{\theta\theta}$ is determined. $R_{x\theta}$ and $R_{y\theta}$ are
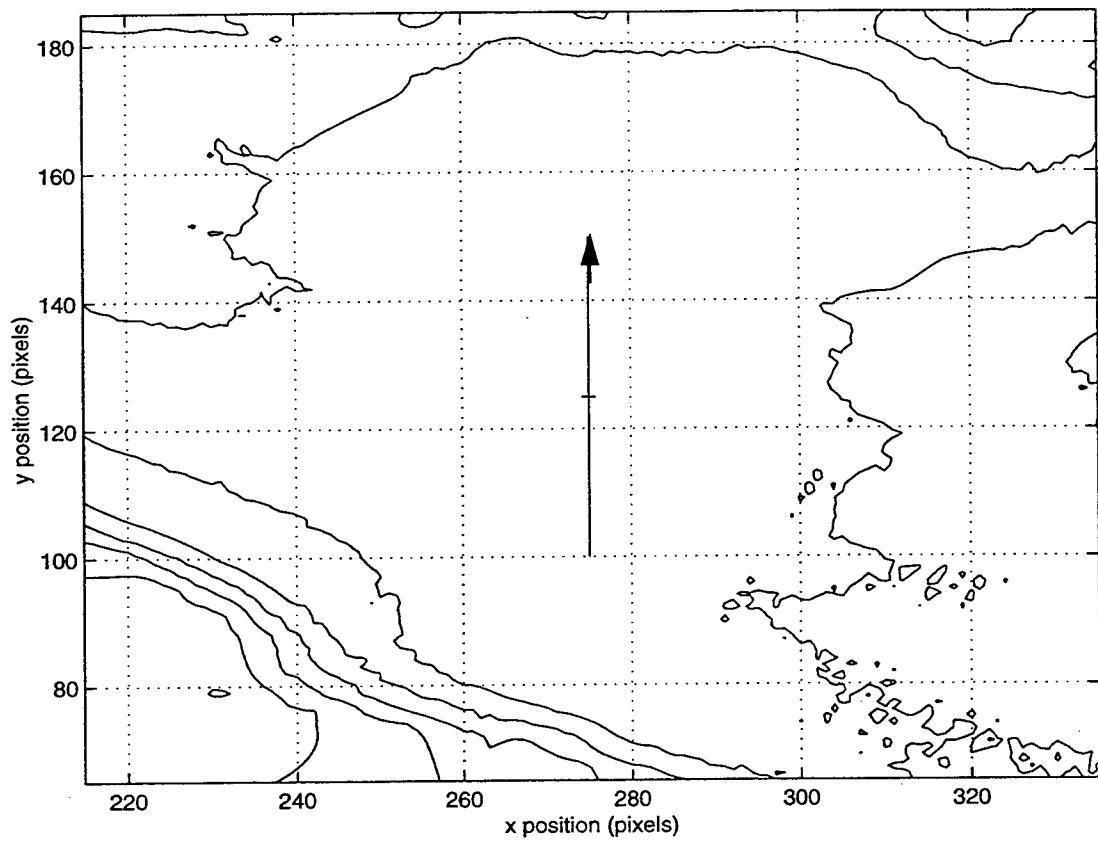
**Figure 4.11.** Map Region about Profile 1.
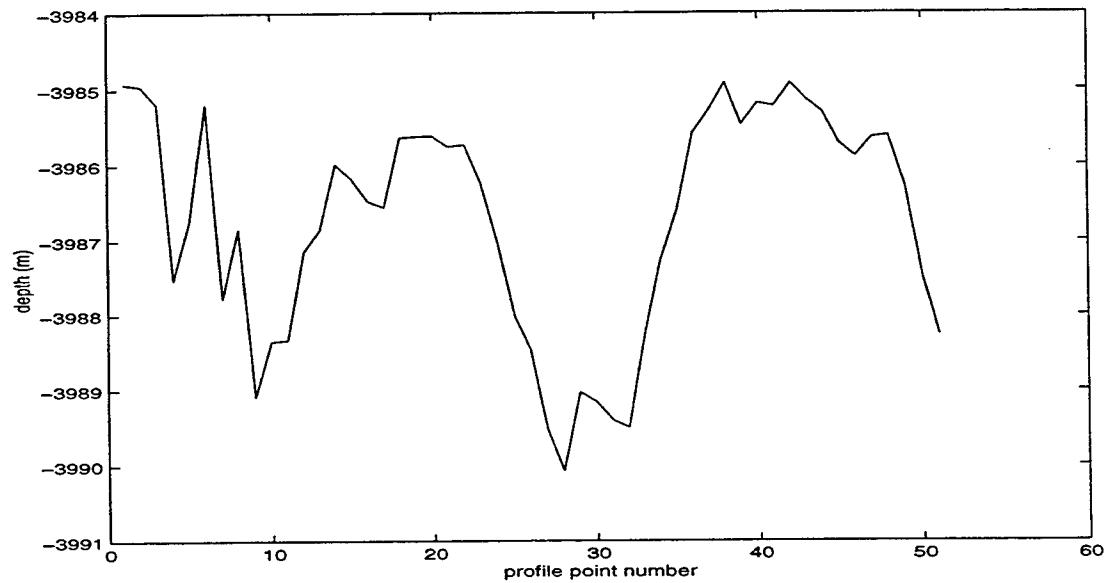
Figure 4.12. Profile 1.

85
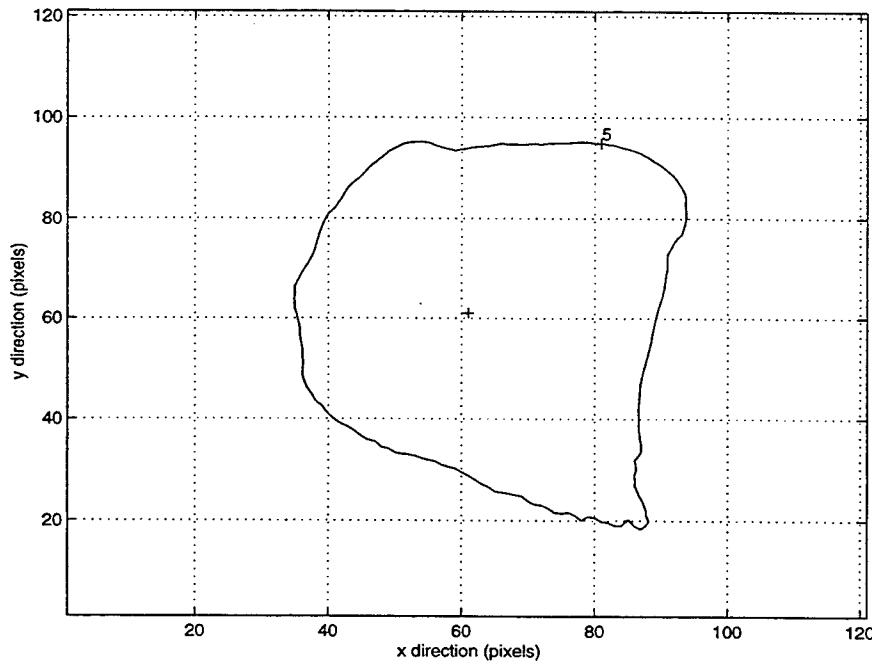
Figure 4.13. XY Contour at *d*=5 for Profile 1.
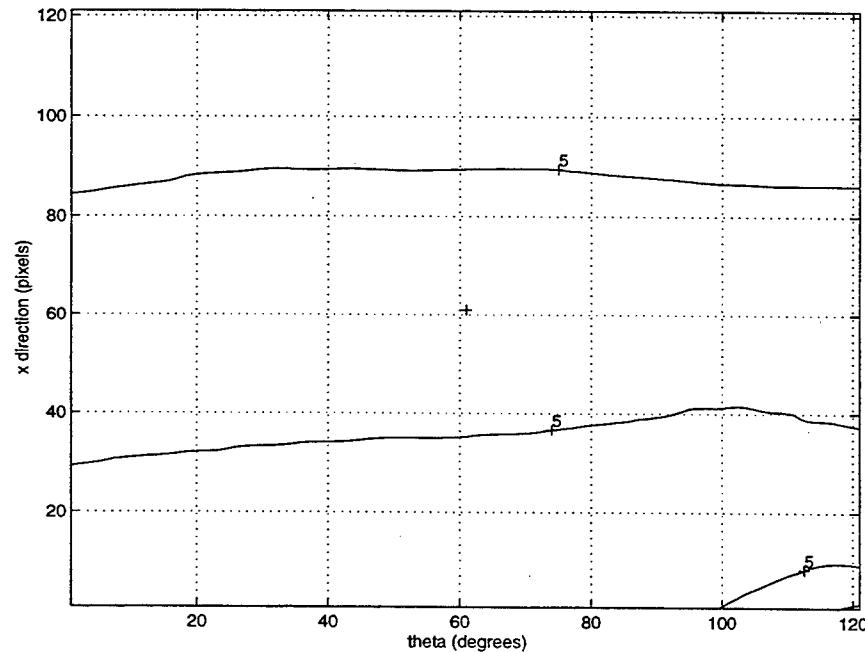
**Figure 4.14.** X-Theta Contour at *d*=5 for Profile 1.

approximated as zero. Finally, the measurement noise covariance matrix for the first

profile, with *d*=5, is

$$R^1 = \begin{bmatrix} 80 & 93 & 0 \\ 93 & 108 & 0 \\ 0 & 0 & 360 \end{bmatrix}.$$ (4.21)
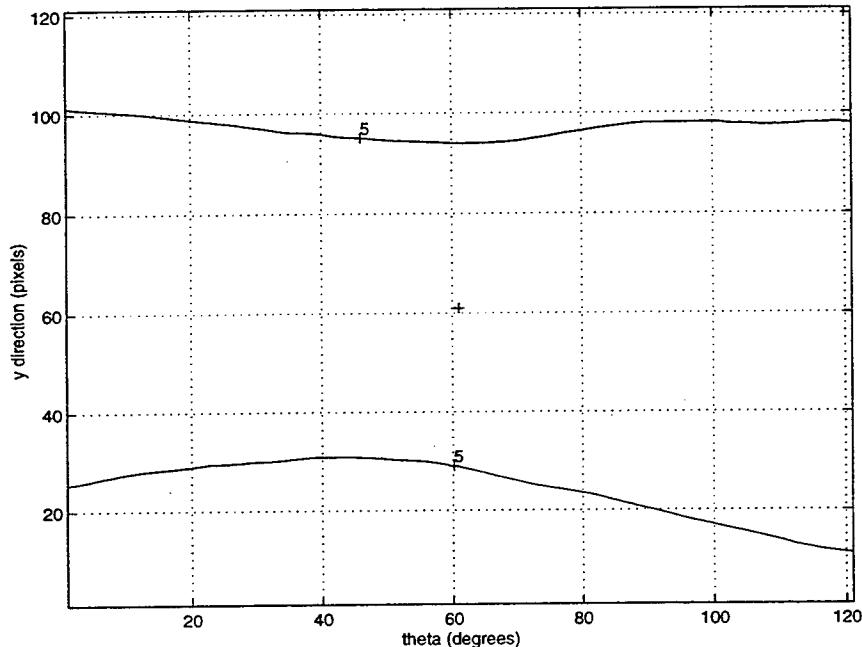
**Figure 4.15.** Y-Theta Contour at $d=5$ for Profile 1

Figure 4.16 shows an enlargement of the map for the neighborhood around the second

profile, which is shown in Figure 4.17. This region has more variation than Region 1 but

consists mostly of gradual slopes. The XY, $X\Theta$, and $Y\Theta$ contours at $d=5$ are shown in

Figures 4.18-4.20. Here again the XY contour follows the terrain, but in this case it does

not close, indicating that the profile matches well as it translates along the ridge. In this

case, it is possible to obtain X_1 and Y_1 but not true values for X_{xy} and Y_{xy}. The

maximum distance point is approximated as the furthest point from the true location that is
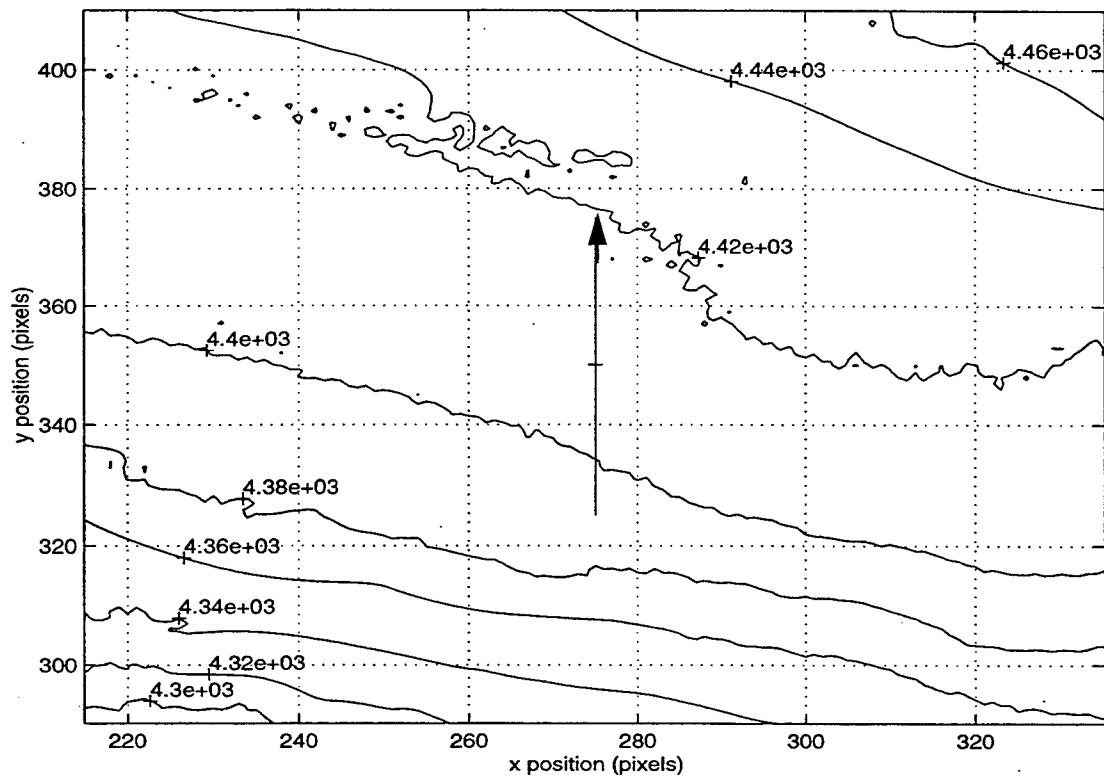
Figure 4.16. Map Region about Profile 2.
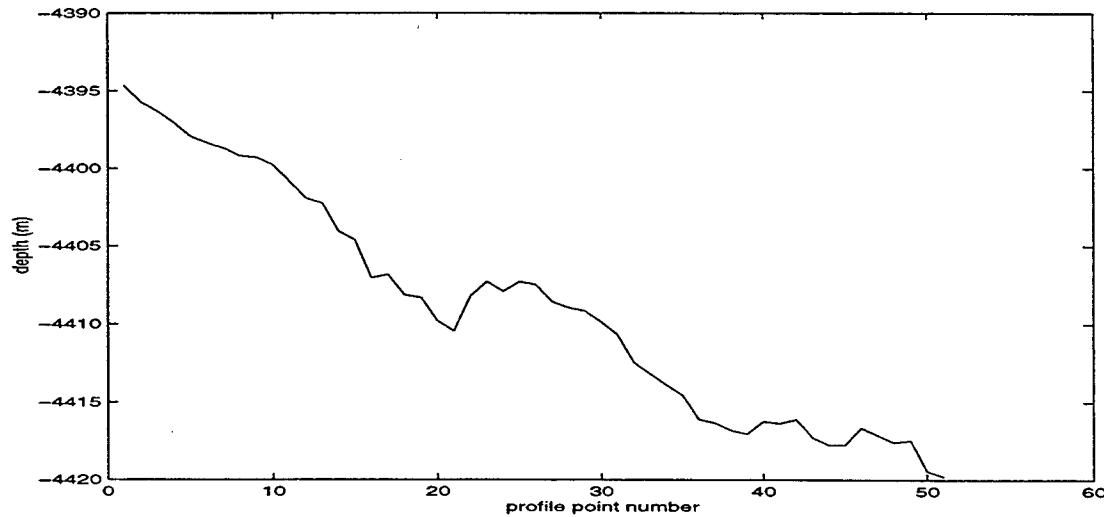
Figure 4.17. Profile 2.

also part of that contour coming closest to the true location. For Figure 4.18, the

approximated (X_{xy}, Y_{xy}) point is indicated.
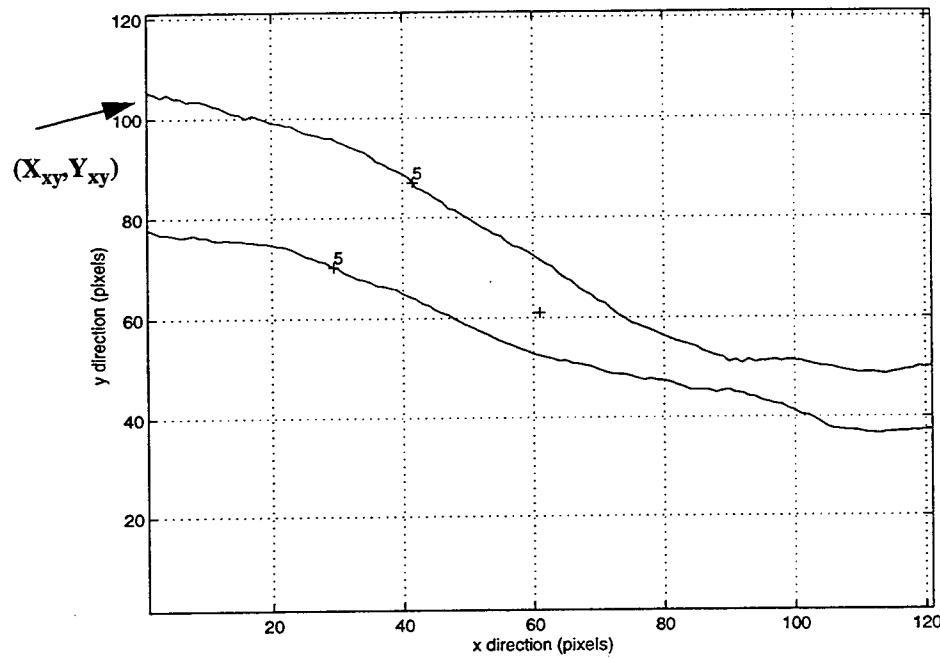
**Figure 4.18.** XY Contour at *d*=5 for Profile 2.
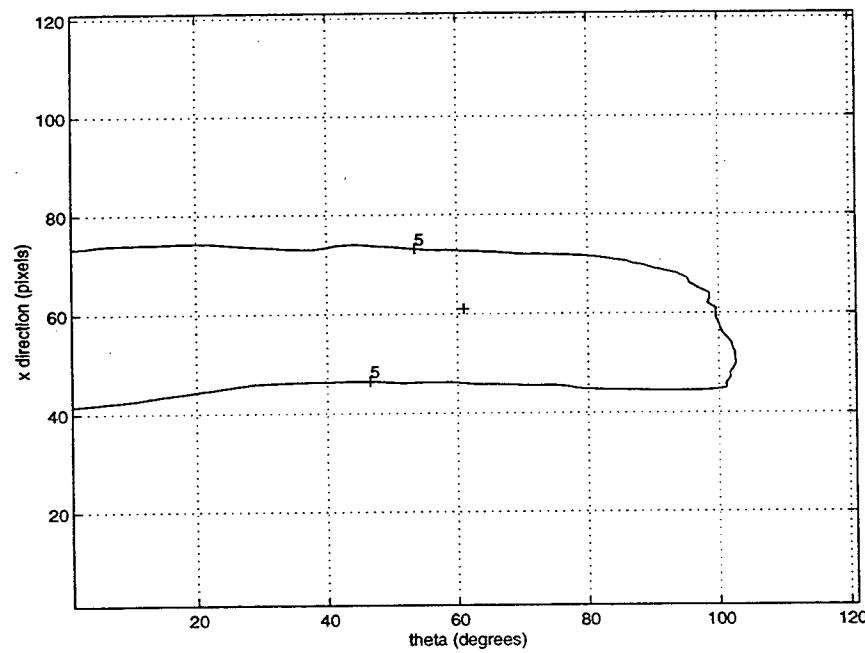

Figure 4.19. X-Theta Contour at *d*=5 for Profile 2.
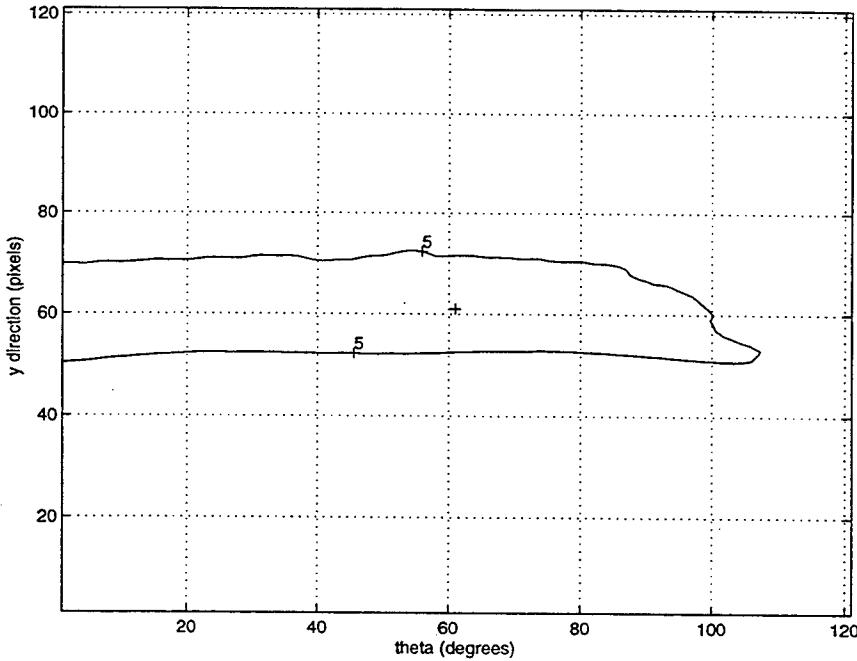
Figure 4.20. Y-Theta Contour at $d=5$ for Profile 2.

The $X\Theta$ and $Y\Theta$ contours for this case close on one side only, which means that rotation in one direction produces large DPS values but that rotation in the opposite direction does not. To keep the procedure consistent, the larger value of $|\theta_1 - \theta_0|$ and $|\theta_2 - \theta_0|$ is chosen, which in this case is the maximum possible value. It is again assumed that the $X\Theta$ and $Y\Theta$ cross terms are zero. Finally, the measurement noise covariance matrix for the second profile is

$$R^2 = \begin{bmatrix} 22 & 14 & 0 \\ 14 & 11 & 0 \\ 0 & 0 & 360 \end{bmatrix}.$$ (4.22)

Figure 4.21 shows an enlargement of the map for the neighborhood around the third profile, which is shown in Figure 4.22. This region is the steepest of the three locations
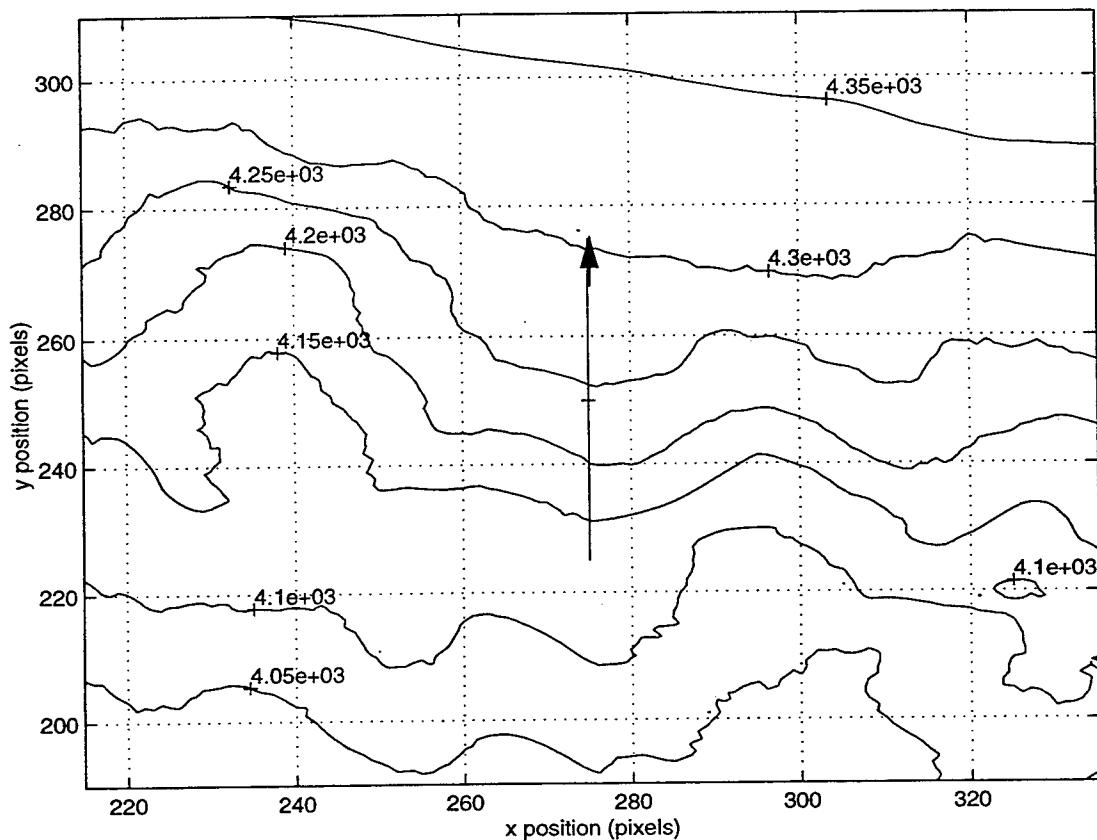
Figure 4.21. Map Region about Profile 3.
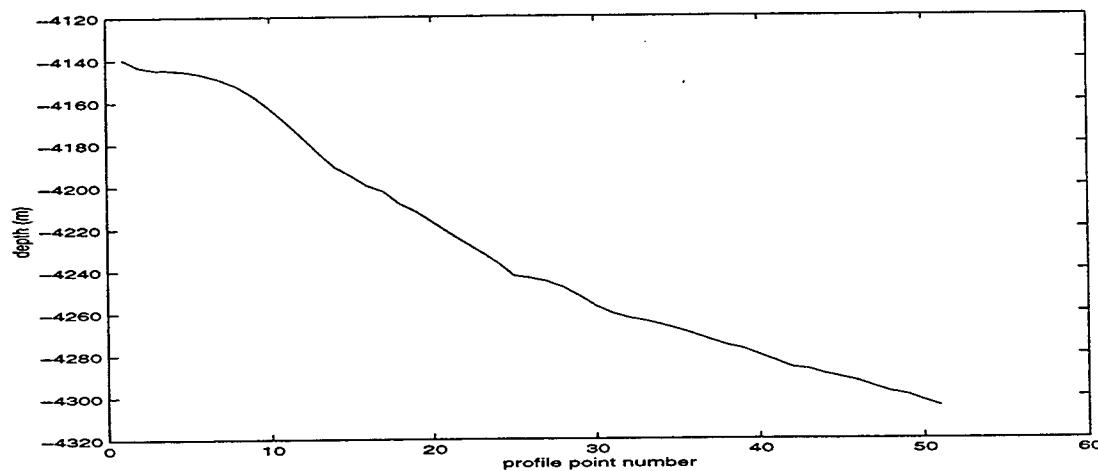
Figure 4.22. Profile 3.

and has the most varying topography. The XY, $X\Theta$, and $Y\Theta$ contours at $d=5$ are shown in

Figures 4.23 -4.25. Here, all contours close about the center point, and thus all values are

calculated according to the normal process. This gives

91
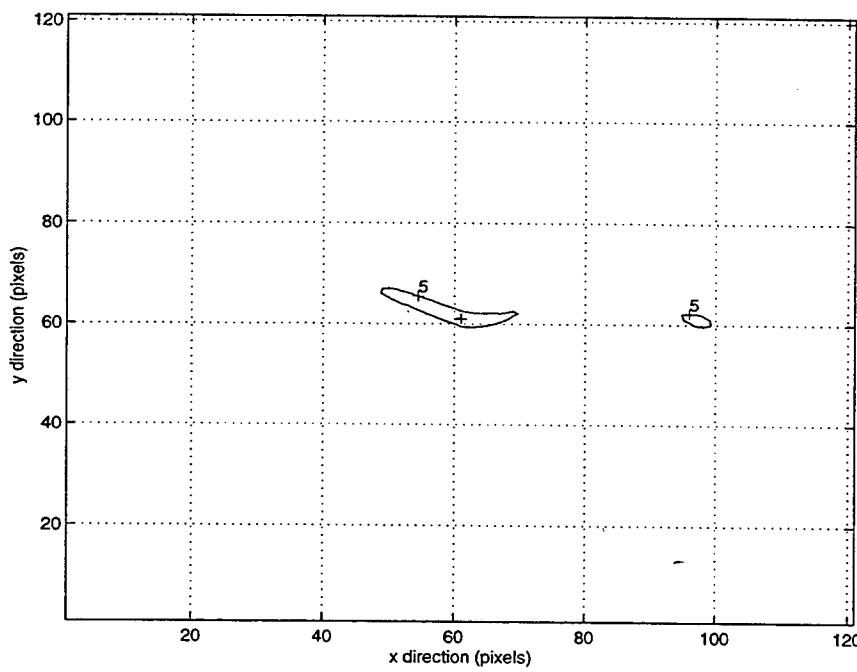
**Figure 4.23.** XY Contour at d=5 for Profile 3.
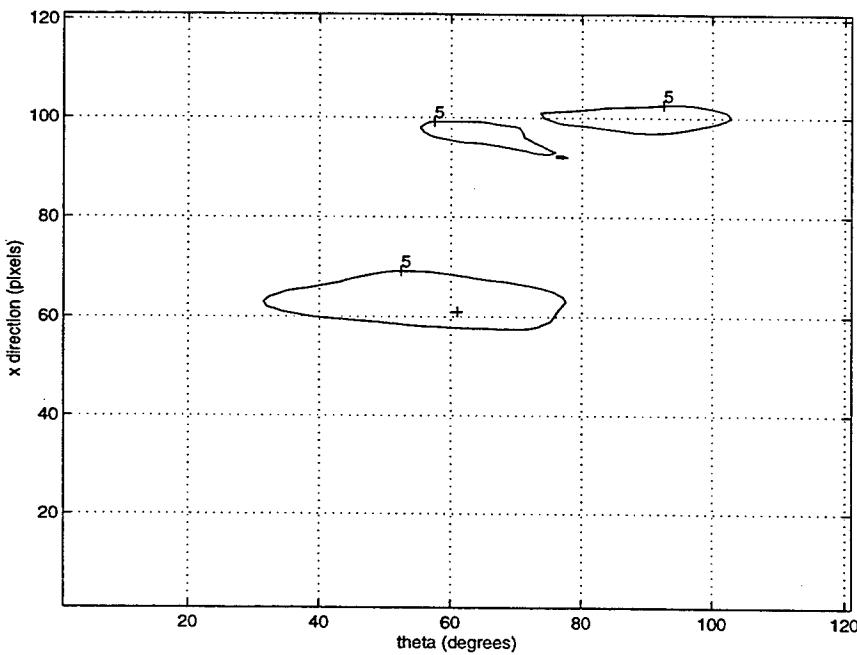
Figure 4.24. X-Theta Contour at d=5 for Profile 3.
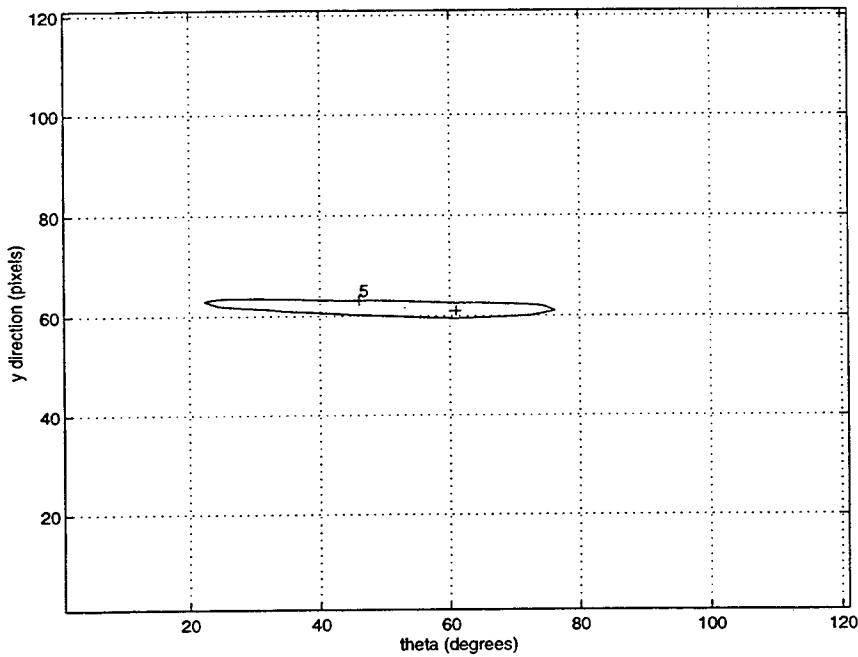
Figure 4.25. Y-Theta Contour at $d=5$ for Profile 3.

$$R^3 = \begin{bmatrix} 5.3 & 0.86 & 38 \\ 0.86 & 0.24 & 5.5 \\ 38 & 5.5 & 71 \end{bmatrix}.$$ (4.23)

However, in this case, all three cross-covariances exceed the acceptable values, so they are

set to the appropriate maxima which give

$$R^3 = \begin{bmatrix} 5.3 & 0.36 & 19 \\ 0.36 & 0.24 & 4.1 \\ 19 & 4.1 & 71 \end{bmatrix}.$$ (4.24)

Interesting to note is that for this case there are multiple closed contours in Figures

4.23 and 4.24. This indicates that the DPS has multiple minima, although the global

minimum occurs at the center point. The presence of alternative minima means that the

exact profile matches well in multiple but separate locations, and thus it is possible that

when the *noisy* profile is matched it matches best at one of these alternative locations. This

further justifies the need to consider multiple matches (see Section 3.2). Several $X\Theta$

contours for multiple values of d are shown in Figure 4.26. An area of large error separates
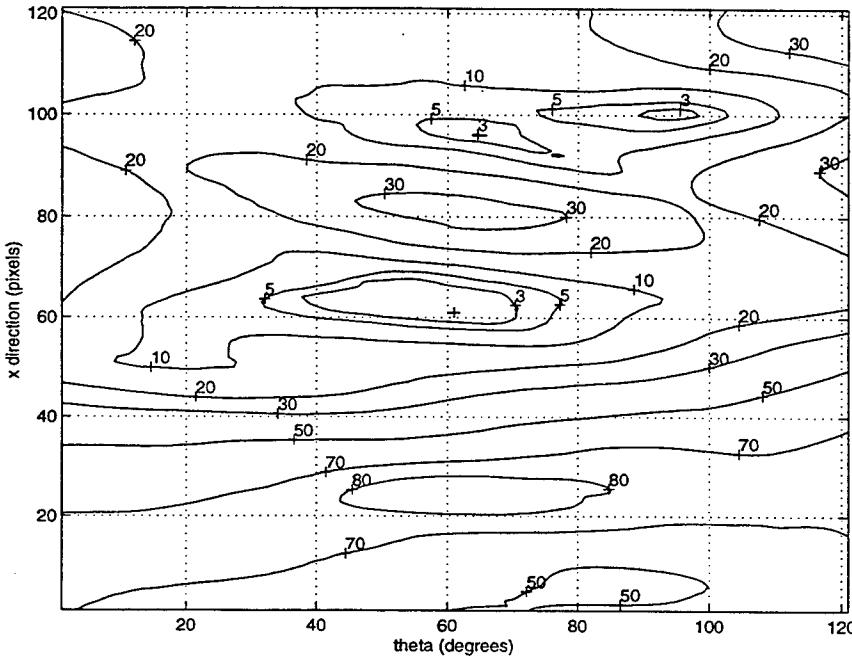
Figure 4.26. Multiple X-Theta Contours for Profile 3.

the two local minima from the global one, and these minima have DPS values of less than

3, which is the level of uncertainty one could expect from the sensors used aboard an AUV

(see Section 5.1).

4.4 Summary

In this chapter, I present the interrelationships between the matching and updating

algorithms of the previous chapter. Specifically, I show that the search range is determined

by the innovation covariance and the amount of downsampling is subsequently determined

by the size of the search range. The probability of detection and the spatial density of false

alarms are determined by the uncertainties on both the map and the current profile and the actual mean absolute differences of the validated measurements. The measurement noise covariance matrix is determined empirically a priori by finding three dissimilarity parameter surfaces for each of the XY, $X\Theta$, and $Y\Theta$ planes for the neighborhood about the true profile location. An elliptic paraboloid is then fit to each dissimilarity parameter surface and the inverses of the second derivatives are taken in a fashion similar to the procedure for determining the Cramer-Rao Lower Bound. In the event that the calculated cross-covariance values are outside the limits set by the correlation coefficient, the appropriate minimum or maximum value is substituted so a valid measurement noise covariance matrix is obtained. Examples show that good matches can be obtained in multiple locations, which verifies that multiple matches should be considered.

Chapter 5

Simulations

The terrain-relative navigation methodology, as described in the previous two chapters, is both sensor and scale independent. It can operate with any point spacing and any uncertainty and need not be modified for each application. Only the sonar pre-processing, which provides the profile and map, is sensor dependent. For this reason, the terrain-relative navigation algorithms are constructed in terms of the pixel spacing instead of specific dimensions.

Presented in this chapter are five tracklines on real deep-ocean bathymetry (see Figure 4.10). Because excessive computations are required to determine the measurement noise covariance matrix when the profile orientation does not align with the axes, the tracklines were selected so that such computational issues could be avoided. The tracklines cover several types of terrain, including a steep ridge, a flat plain, and a gradually sloping area. After a brief discussion of system parameters, the navigation results are first displayed in pixel units for a qualitative understanding then discussed quantitatively in true units (meters).

5.1 Parameter Selection

The parameters that remain constant for all simulations are summarized in Table 5.1.

For this demonstration, the state is measured directly. The governing dynamic equation is

$$x(k+1) = x(k) + u(k+1)\Delta k. \tag{5.1}$$

Table 5.1 Constant Simulation Parameters

F(k)=I_3	$R_{x\theta}$=0	σ_m=1m	g=4
G(k)=I_3	$R_{y\theta}$=0	σ_p=3m	percent threshold=10%
H(k)=I_3	$R_{\theta\theta}$=750 degrees^2	d=2.4	profile length =50

The matrices, *F(k)*, *G(k)*, and *H(k+1)* are all 3x3 identity matrices, so the state and measurement vectors are both

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix},$$

and the control inputs are the derivatives

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}.$$

The variable Δk represents the amount of time between updates. For a vehicle traveling at a speed of 1 m/s, 5 s are required for the vehicle location to change by one pixel, since the pixel spacing is 5 m. Because I do not expect sub-pixel resolution for the navigation, Δk is chosen to represent 10 s, so that a vehicle with velocity of 1 m/s (which is typical of current AUVs) will travel approximately two pixels in a time step. So $\Delta t = 10$ s, but $\Delta k = 1$.

The process noise covariance matrix $Q(k)$ (see Section 3.2.1) is found from [Leader] and [RD Instruments] by assuming that internal navigation comes from an on-board gyrocompass and Doppler velocimeter log, such as the Workhorse Navigator. The values should of course grow with time. However, given that Δt is 10 s for a single time step Δk, and if cross-covariances are assumed zero,

$$Q(k) = \begin{bmatrix} 0.36 & 0 & 0 \\ 0 & 0.36 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}, \tag{5.2}$$

where the x and y terms are in units of pixel^2, and the θ term is in degrees^2.

The simulations in this chapter are semi-aided by the Doppler log. The control input u is the velocity vector plus noise, which drives the internal navigation. Strictly speaking, each velocity measurement should be another entry in the measurement vector and could be another entry in the state vector. For simplicity in demonstrating the independent capabilities of the terrain-relative navigation however, only the output of the terrain-relative navigation matching process defines the measurement vector. So, the values for the control input u and the process noise covariance matrix $Q(k)$ are in part determined by the Doppler log.

The profile consists of 50 depth values, equally spaced, with the same spacing as the pixels of the map (5 m), and the depth uncertainty on the profile points is constant. Current high-quality sensors can provide a sonar scan with depth uncertainty of roughly 2-5 m [Stewart 1997]. I choose $\sigma_p = 3$ m as a representative value. A larger value influences

the calculations of P_D, λ, and $R(k+1)$ (see Section 4.3). Also, although most parameters are normalized over the number of points in the profile, generally, longer profiles contain more information and are thus harder to match well, because the additional information makes the profile more unique. Shorter profiles contain less information and can be matched well more easily. This is implicit in the terrain-relative navigation process, and varying the profile length affects the number of matches as well as the values of E and F (see Section 4.2). However, for the simulations of this research, the profile length is always 50 points. This number was selected because a profile of this length provides sufficient information about the local topography without demanding excessive computational resources. The effects of other parameters have been discussed elsewhere, but I state here that I assume the map is fairly accurate with depth uncertainty of $\sigma_m = 1$ m (see Section 4.2); the gate extends to four deviations, or $g=4$ (see Section 3.2.2.1 and 4.1); and the percentage threshold is 10% (see Sections 3.1 and 4.2).

As stated above, in the field an AUV would be equipped with a fairly accurate compass or angular velocimeter. Consequently, the orientation measurements provided by the terrain-relative navigation have much higher uncertainties than the on-board internal navigational sensors. Although I have shown that terrain-relative navigation can provide profile orientation measurements and the associated uncertainties (see Chapter 3 and Section 4.3), for the demonstration in this chapter I assume that compass or velocimeter measurements are available and much more trustworthy. Thus, I set $R_{\theta\theta}$ to its maximum value and decouple the orientation cross-covariances by setting the values of $R_{x\theta}$ and $R_{y\theta}$ to zero. This minimizes the effect of the terrain-relative navigation orientation

measurements. A random variable with deviation of unity has an average value of approximately 0.8. Thus, profiles with uncertainty of $\sigma_p = 1$ will give a threshold of d=0.8, and, extrapolating to my case of $\sigma_p = 3$, d=2.4. So, for all $R(k+1)$ calculations, d=2.4, and the maximum value for the measurement noise orientation uncertainty is (see Equation 4.15)

$$R_{\theta\theta} = \frac{60^2}{2(2.4)} = 750.$$

(5.3)

in units of degrees^2.

5.2 Tracklines

Five tracklines across the deep-ocean bathymetry of Figure 4.10 were tested to determine the performance of the terrain-relative navigation system. This section presents those tracklines with a qualitative analysis of the results.

The first trackline has initial conditions of both the true and estimated states as

$$x = \begin{bmatrix} 300 \\ 100 \\ 0 \end{bmatrix},$$

(5.4)

with a state covariance matrix

$$P = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}.$$

(5.5)

The control input vector is

$$u = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix} + n, \tag{5.6}$$

where n is a Gaussian white noise vector with covariance 0.6 $(m/s)^2$ for the x and y

velocities and 0.1 $(degrees/sec)^2$ for the angular velocity. So, the vehicle travels parallel to

the y axis, perpendicular to the profile orientation. The true trackline continues for 150

time steps and ends at

$$x = \begin{bmatrix} 300 \\ 400 \\ 0 \end{bmatrix}. \tag{5.7}$$

Figure 5.1 shows an enlargement of the map for the region in the neighborhood of

Trackline 1. Figures 5.2 and 5.3 show the values of R_{xx} and R_{yy} for Region 1 with this

profile orientation. The vehicle starts in a relatively flat area with large measurement

uncertainty, particularly in the y direction. It then passes through steep terrain, which

provides very accurate navigation, as both R_{xx} and R_{yy} are small. Towards the end of the

trackline, the terrain has less variability and again gives large measurement uncertainty,

but this time more so in the x direction.

Figures 5.4 and 5.5 show the true and navigated tracks and the associated covariances

for Trackline 1. At the beginning of the trackline, the estimated state does not track the

true state well and the covariances stay large. As the vehicle moves into the steep terrain,
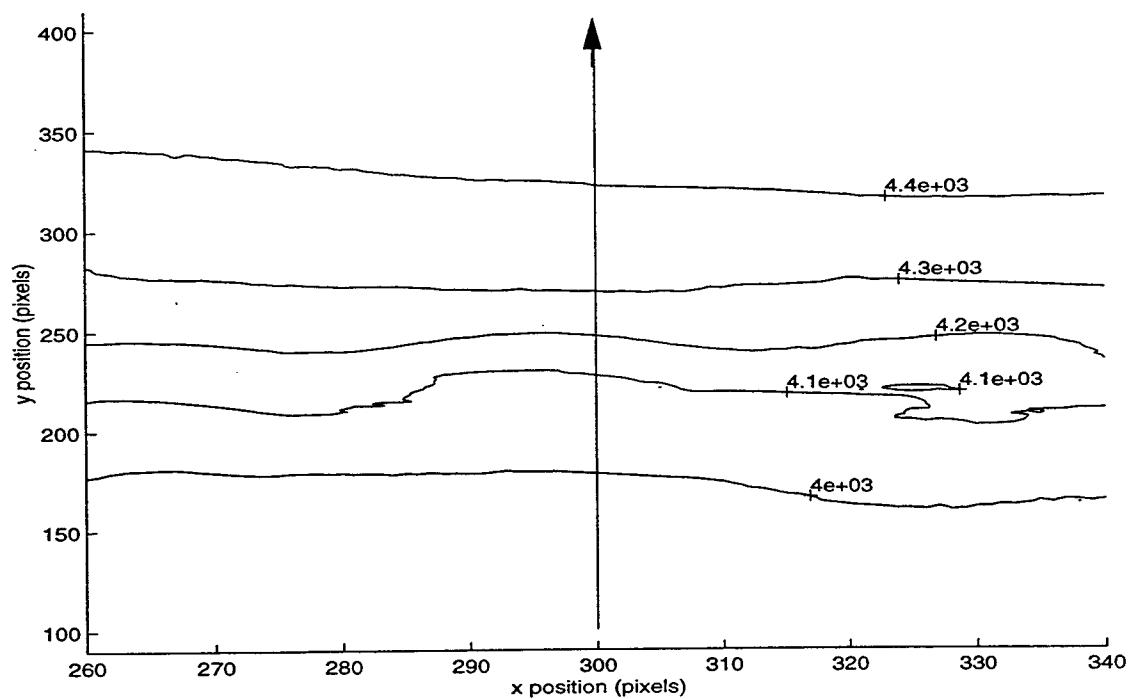
Figure 5.1. Region 1 with Trackline 1.
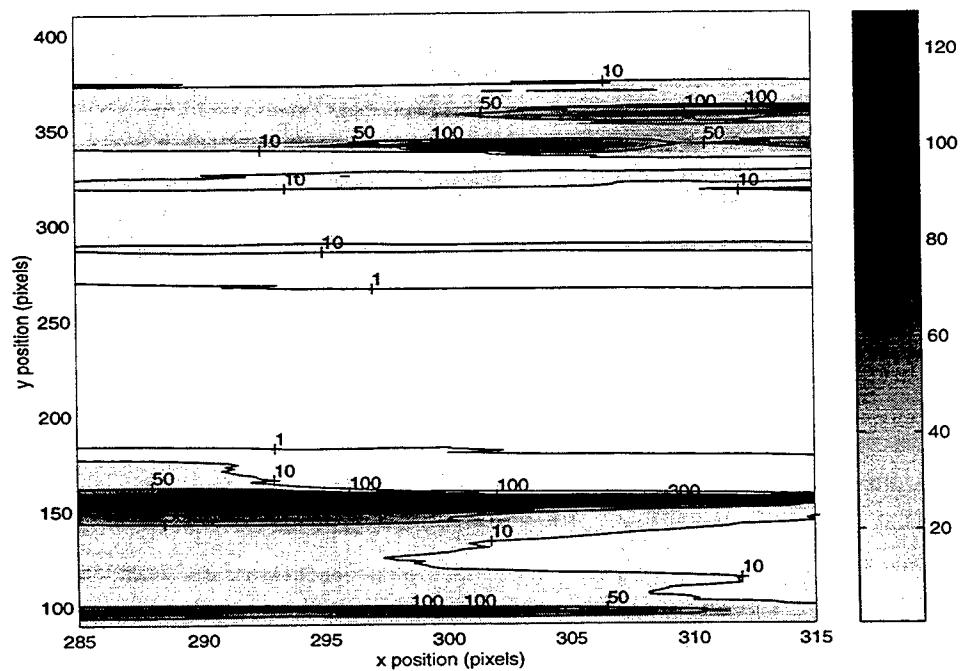
Figure 5.2. R_{xx} for Trackline 1.

103
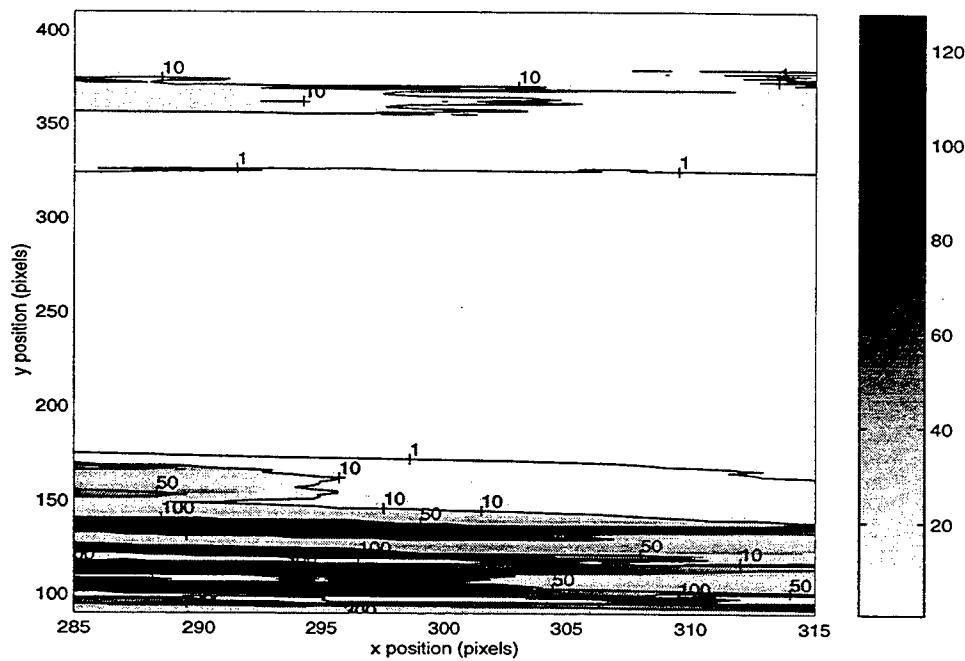
Figure 5.3. R_{yy} for Trackline 1.
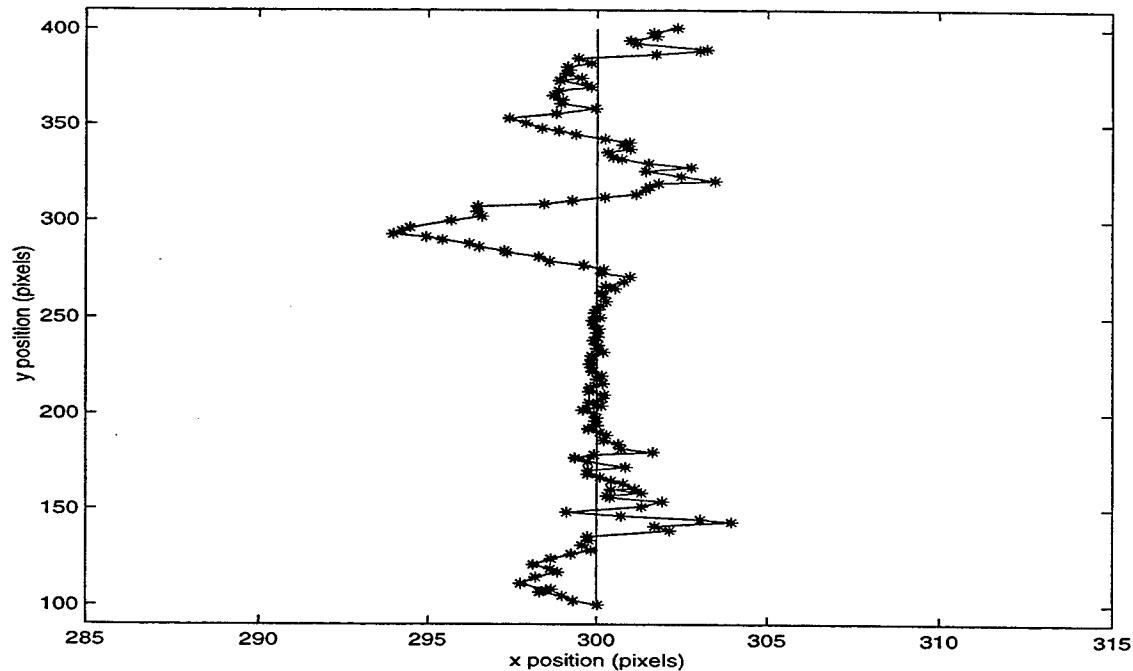
Figure 5.4. True (-) and Navigated (*) Trackline 1.

more accurate navigation is achieved, and the estimated state holds the true track fairly

well with small covariance values. Then, the vehicle leaves this area and enters a region

with less variable terrain, and the estimated position again does not track the true state

104
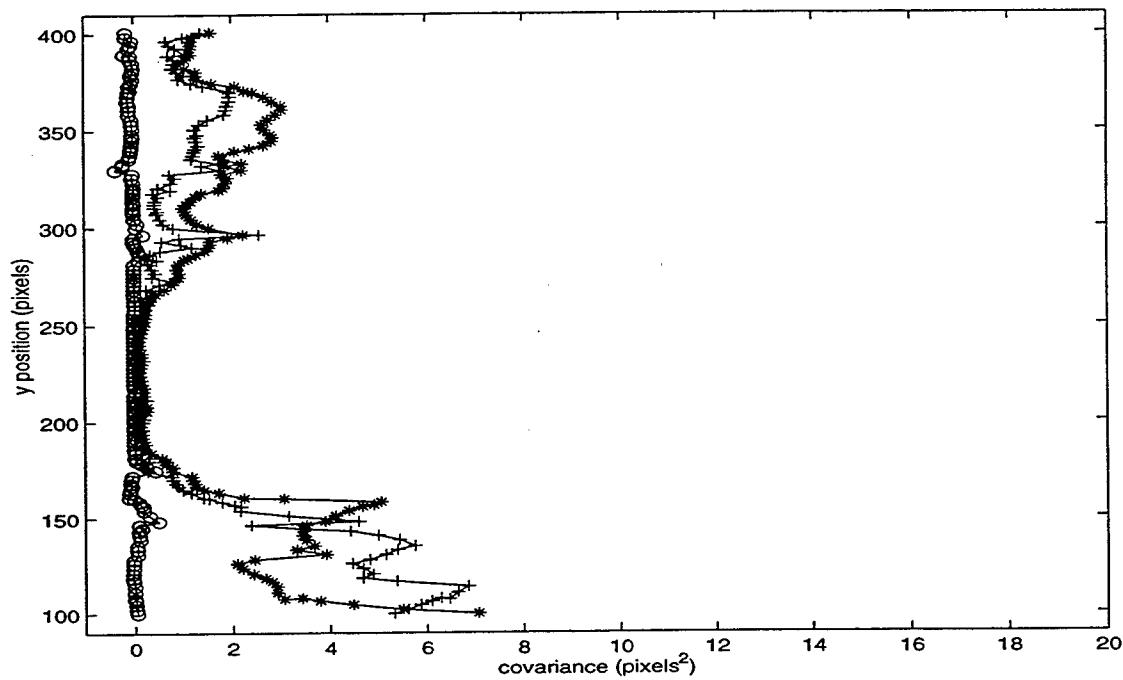
Figure 5.5. Covariances for Trackline 1.

$xx(*)$, $yy(+)$, $xy(o)$.

well. In this difficult area, the state covariance grows as the estimate becomes well off-track. Eventually, the measurements pull the estimate back towards the correct values, so the navigation does not diverge greatly. This is a property of a well-defined Kalman Filter [Brown and Hwang].

For this trackline, I also show, as an example, the Kalman Gain value $W(k+1)$ for the y direction (see Figure 5.6). Note that the gain on the measurements is low in the areas where the terrain does not support good navigation but is high in the area where the terrain should provide accurate navigation. The Kalman Gain is a direct function of the measurement covariance matrix (Equations 3.8 and 3.11), so the gain values reflect the measurement uncertainty.

Trackline 2 is the reverse of Trackline 1, namely it starts at
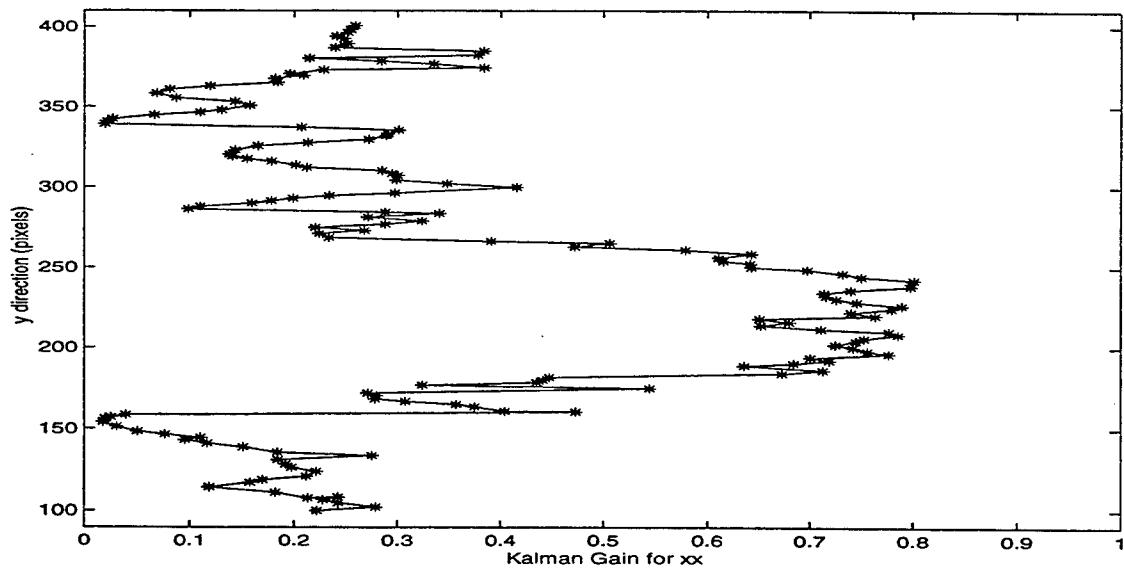
Figure 5.6. Kalman Gain for xx, Trackline 1.

$$x = \begin{bmatrix} 300 \\ 400 \\ 0 \end{bmatrix}, \tag{5.8}$$

has control input of

$$u = \begin{bmatrix} 0 \\ -2 \\ 0 \end{bmatrix} + n, \tag{5.9}$$

where n is as above and heads toward

$$x = \begin{bmatrix} 300 \\ 100 \\ 0 \end{bmatrix}. \tag{5.10}$$

The initial state covariance is the same, and Figures 5.1-5.3 are of course also valid for this

trackline.

Figures 5.7 and 5.8 show the true and navigated tracks and the associated covariances

for Trackline 2; the results are similar to those of Trackline 1. The vehicle does not hold
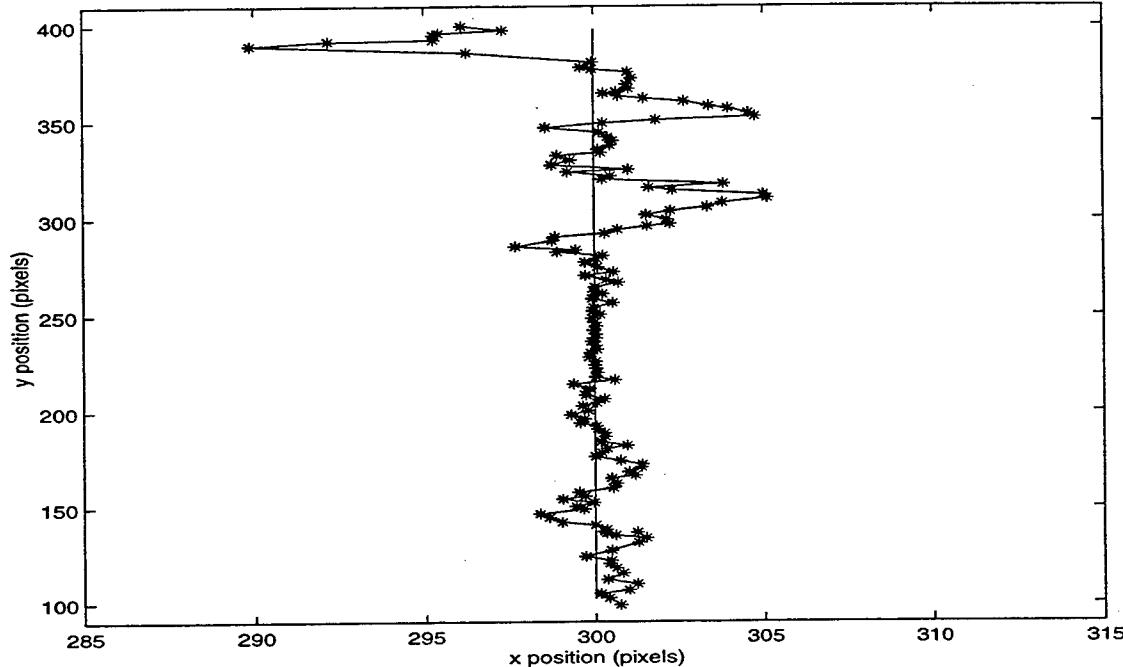

Figure 5.7. True (-) and Navigated (*) Trackline 2.

the track well at the beginning but improves greatly towards the middle and then gets

somewhat off-track towards the end. The covariances also give similar results: large in the

beginning, settling to small values in the middle, and growing again towards the end.

The third trackline has initial conditions of both the true and estimated states as

$$x = \begin{bmatrix} 150 \\ 100 \\ 0 \end{bmatrix},$$ (5.11)

with a state covariance matrix
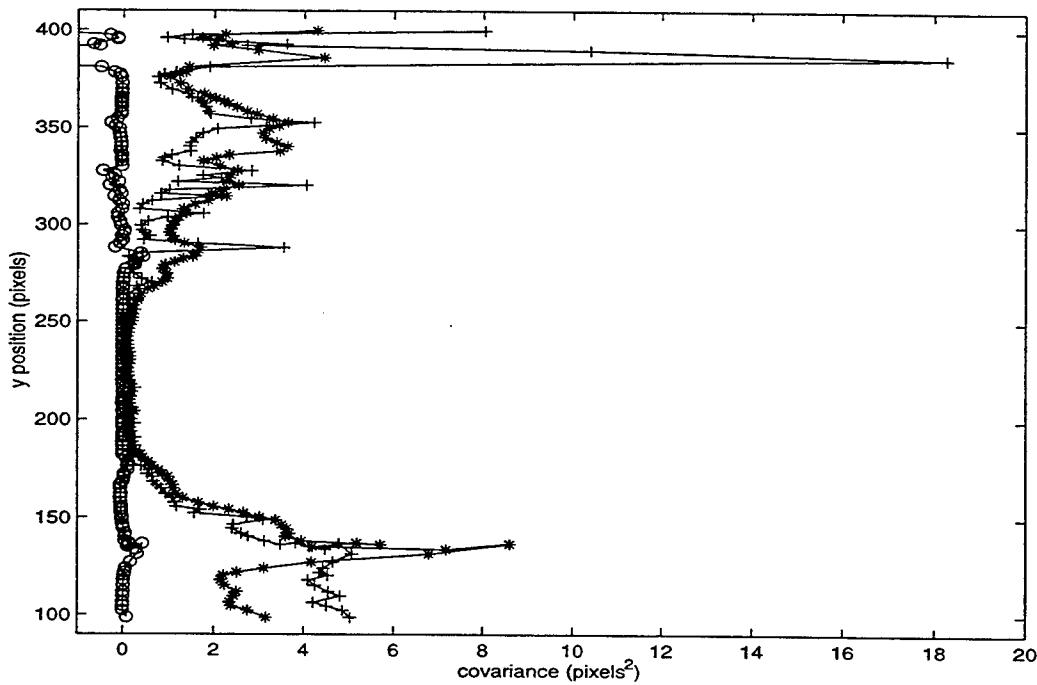
Figure 5.8. Covariances for Trackline 2.

$xx(*)$, $yy(+)$, $xy(\mathrm{o})$.

$$P = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}. \qquad (5.12)$$

The control input vector is the same as that of Trackline 1,

$$u = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix} + n, \qquad (5.13)$$

so Trackline 3 is also parallel to the y axis and ends at

$$x = \begin{bmatrix} 150 \\ 400 \\ 0 \end{bmatrix}, \qquad (5.14)$$

108
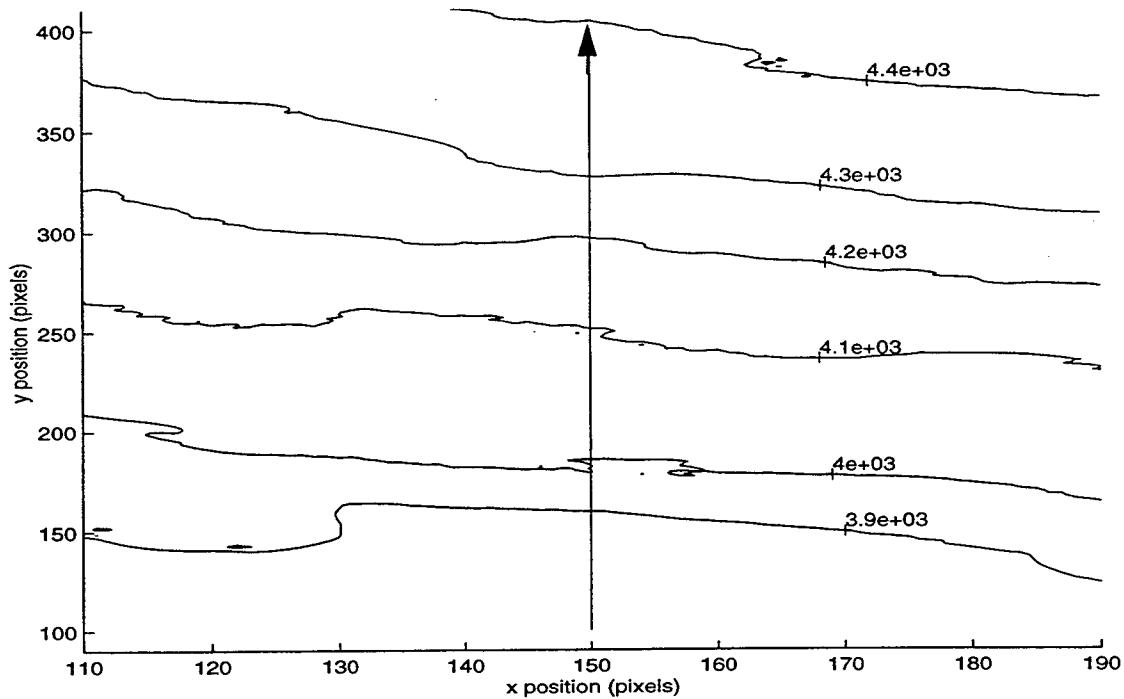
Figure 5.9. Region 3 and Trackline 3.

given 150 time steps. Figure 5.9 shows an enlargement of the map for this region; Figures

5.10 and 5.11 show the corresponding values of R_{xx} and R_{yy}. Again, the vehicle starts in

a relatively flat region with somewhat large (although not as large as Trackline 1)

measurement uncertainty values, especially in the x direction. The vehicle then travels

across the steep ridge. However, in this case it does not exit the area of highly variable

terrain.

Figures 5.12 and 5.13 show the true and navigated tracks and the associated

covariances for this trackline. At the very beginning, the estimated state does not track the

true state well and the covariances are large. However, after a brief period when the

covariances become very large, the tracking is fairly good and the covariances stay small.

Trackline 4 has initial conditions of both the true and estimated states as
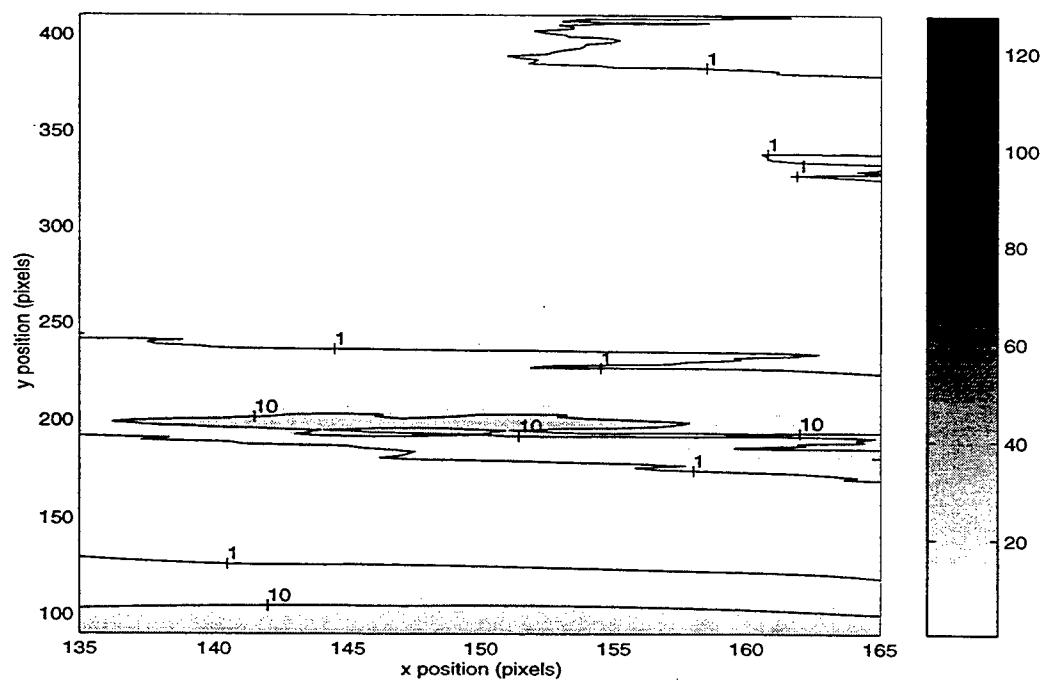
**Figure 5.10.** R_{xx} for Trackline 3.
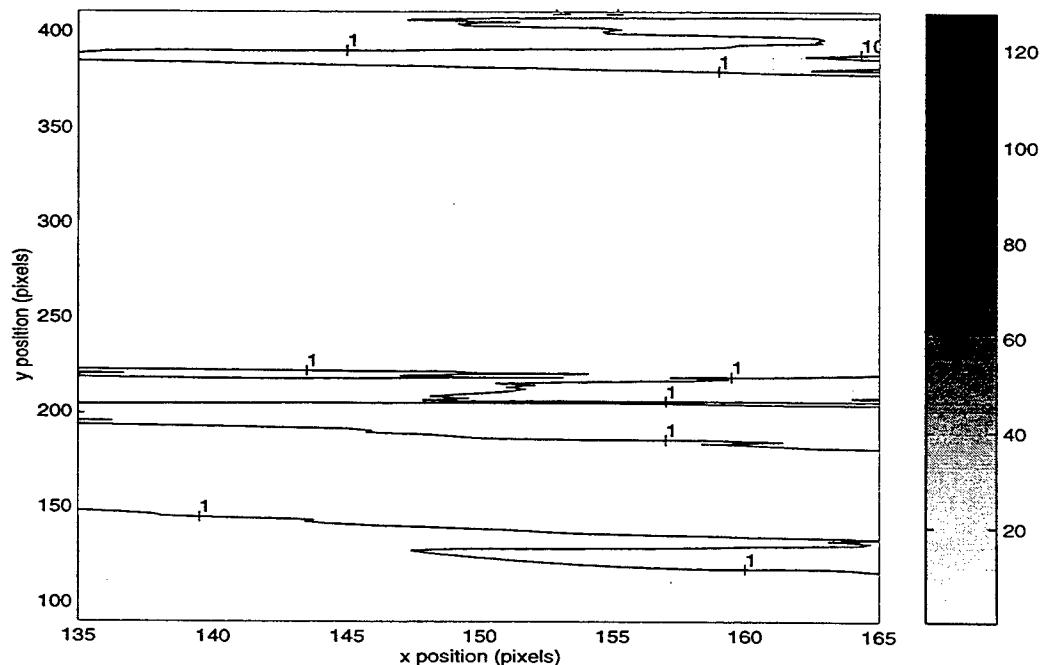
Figure 5.11. R_{yy} for Trackline 3.
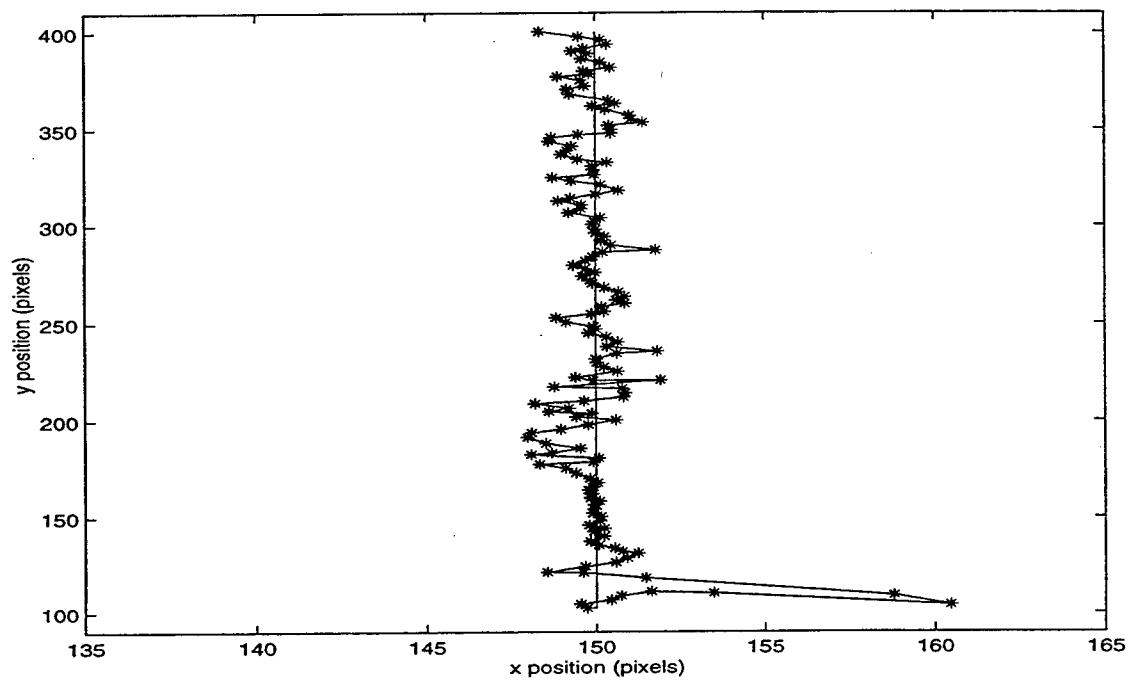
Figure 5.12. True (-) and Navigated (*) Trackline 3.
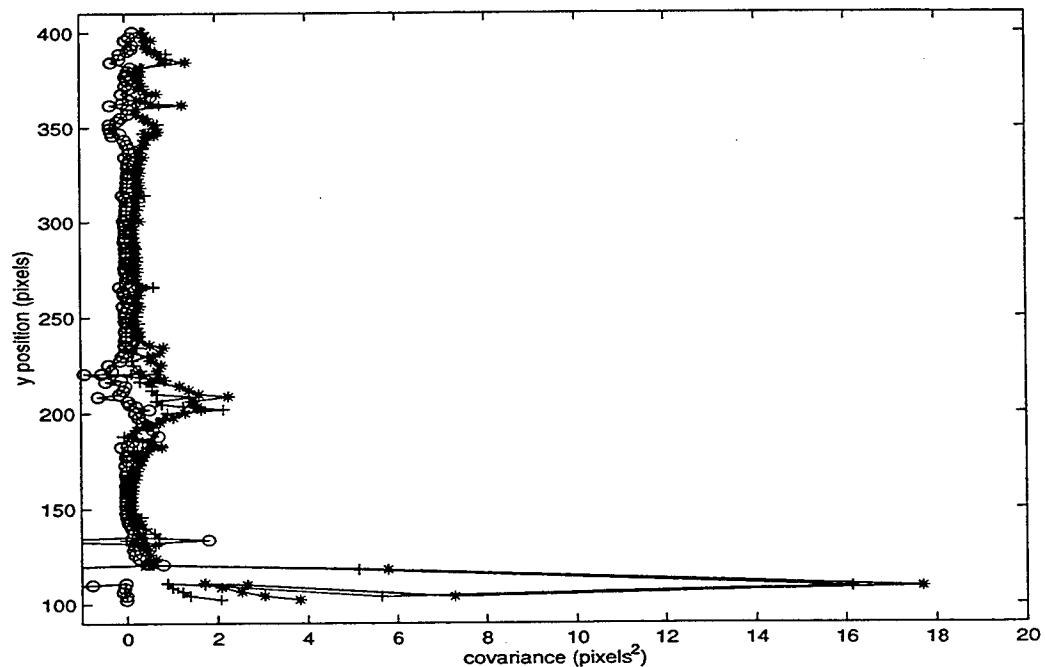

Figure 5.13. Covariances for Trackline 3.

xx(*), yy(+), xy(o).

111

$$x = \begin{bmatrix} 300 \\ 400 \\ 90 \end{bmatrix}, \qquad (5.15)$$

with a state covariance matrix

$$P = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}. \qquad (5.16)$$

The control input vector is

$$u = \begin{bmatrix} -2 \\ 0 \\ 0 \end{bmatrix} + n. \qquad (5.17)$$

This time the vehicle travels parallel to the x axis, and the profiles are parallel to the y axis. The true trackline extends for 100 time steps and ends at

$$x = \begin{bmatrix} 100 \\ 400 \\ 90 \end{bmatrix}. \qquad (5.18)$$

Figure 5.14 shows an enlargement of the map for the region in the neighborhood of Trackline 4. Figures 5.15 and 5.16 show the values of R_{xx} and R_{yy} for Region 4 with this profile orientation. The vehicle starts in the gradually sloped area with large measurement uncertainty, particularly in the x direction, and passes through a region where R_{xx} is very large. Then, it travels to the steep ridge where more accurate navigation is possible as both

112
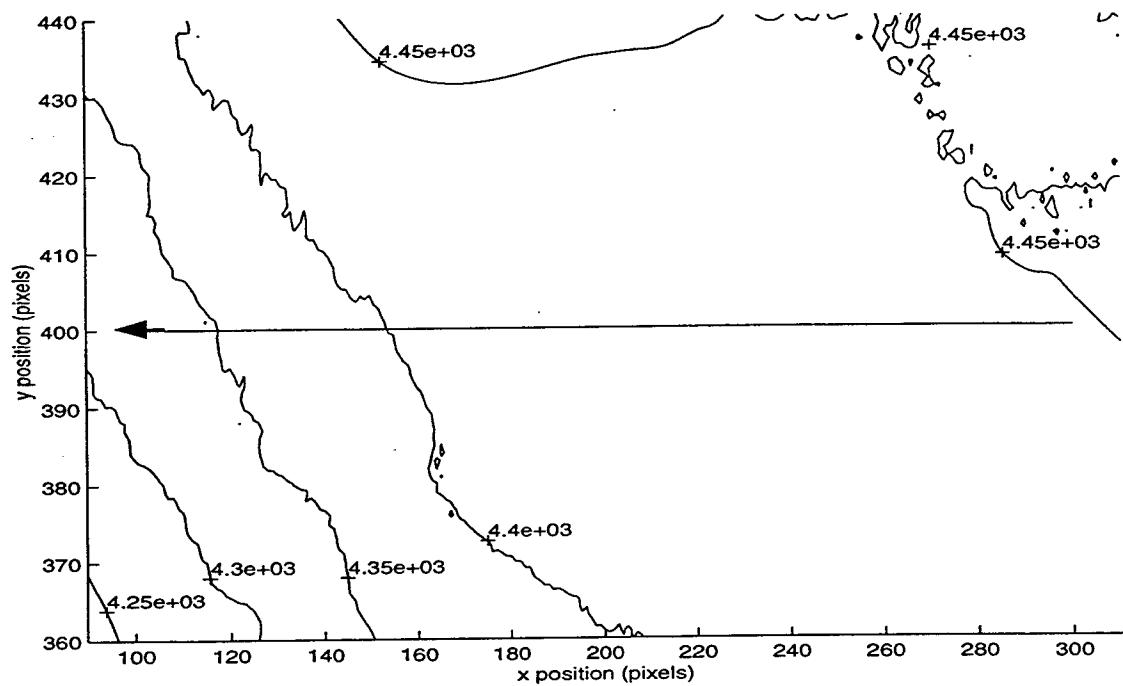
Figure 5.14. Region 4 and Trackline 4.
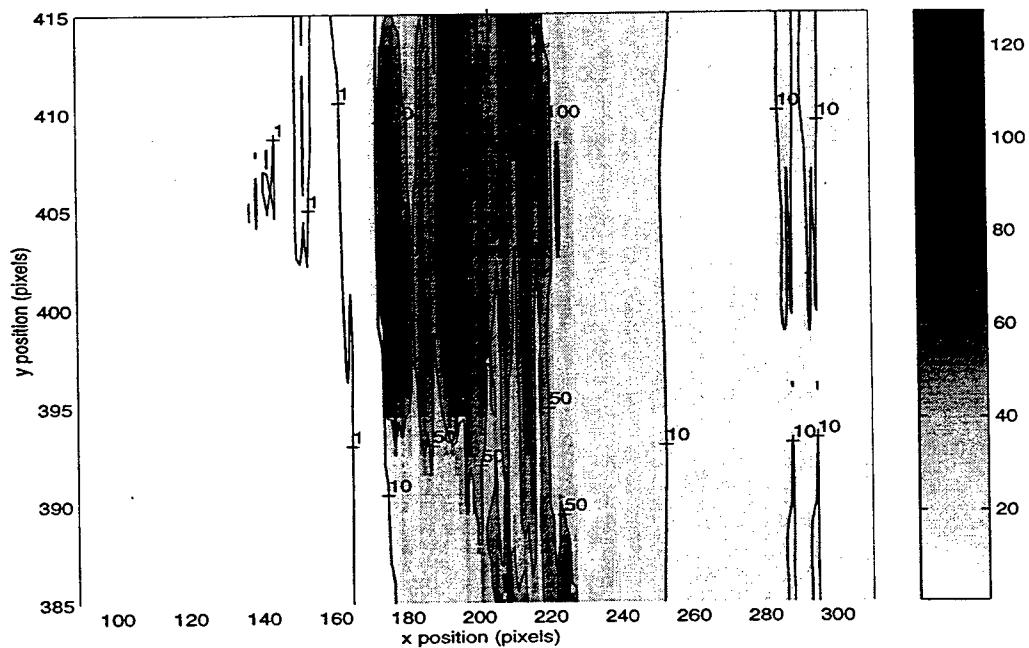
Figure 5.15. R_{xx} for Trackline 4.

113
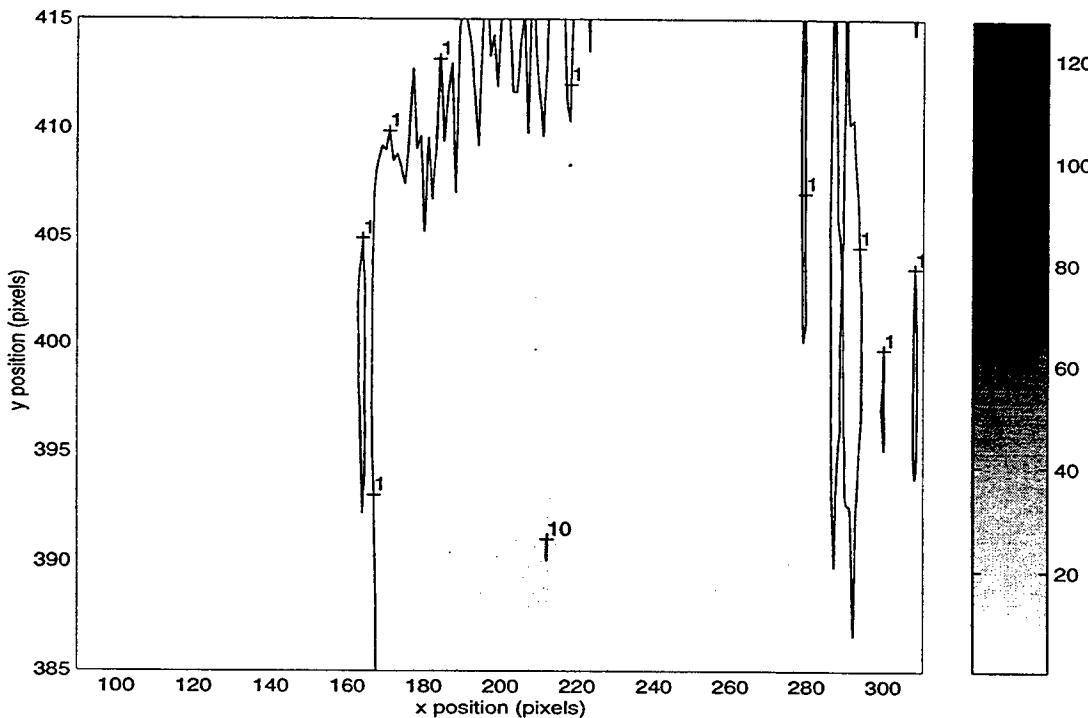
Figure 5.16. R_{yy} for Trackline 4.

R_{xx} and R_{yy} are smaller.

Figures 5.17 and 5.18 show the true and navigated tracks and the associated

covariances for Trackline 4. The navigation is fairly good throughout the trackline.

Although no well-defined transition point appears in the trackline, the covariances

decrease as the vehicle moves over the ridge. This corresponds roughly to when $x=190$.

The final trackline travels directly along the ridge. The initial conditions are

$$x = \begin{bmatrix} 350 \\ 250 \\ 90 \end{bmatrix},$$ (5.19)

with a state covariance matrix

114
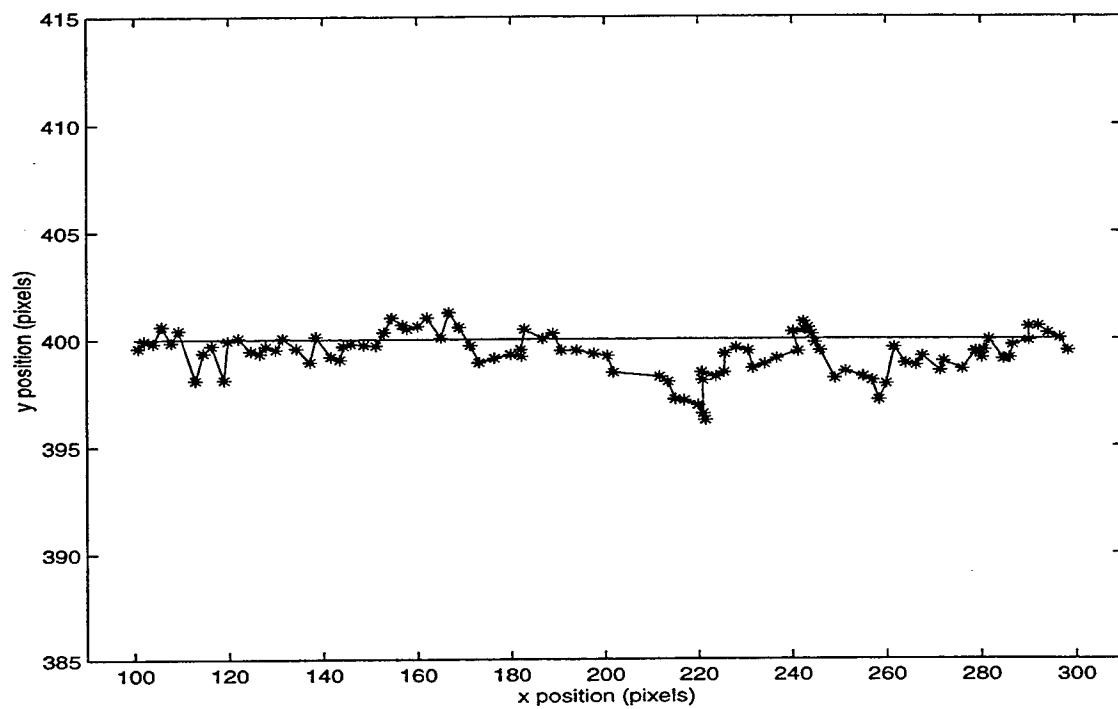
Figure 5.17. True (-) and Navigated (*) Trackline 4.
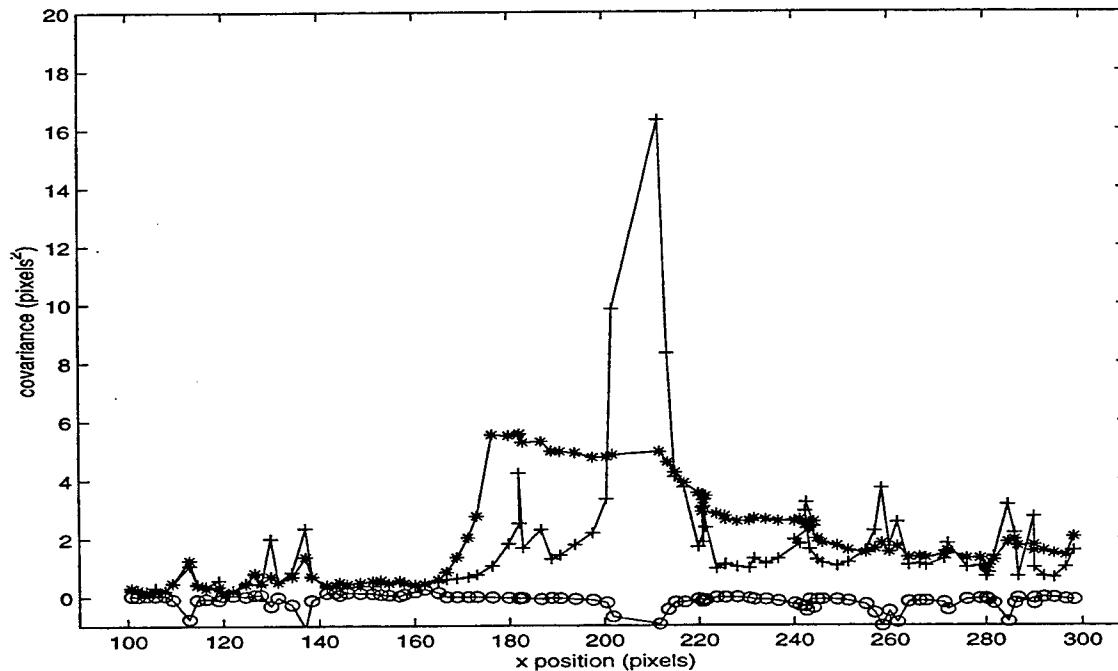

Figure 5.18. Covariances for Trackline 4.

xx(*), yy(+), xy(o).

115

$$P = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}, \tag{5.20}$$

and a control input vector

$$u = \begin{bmatrix} -2 \\ 0 \\ 0 \end{bmatrix} + n. \tag{5.21}$$

Again, the vehicle travels parallel to the x axis for 100 time steps, and the profiles are parallel to the y axis. Figure 5.19 shows an enlargement of the map for the region in the
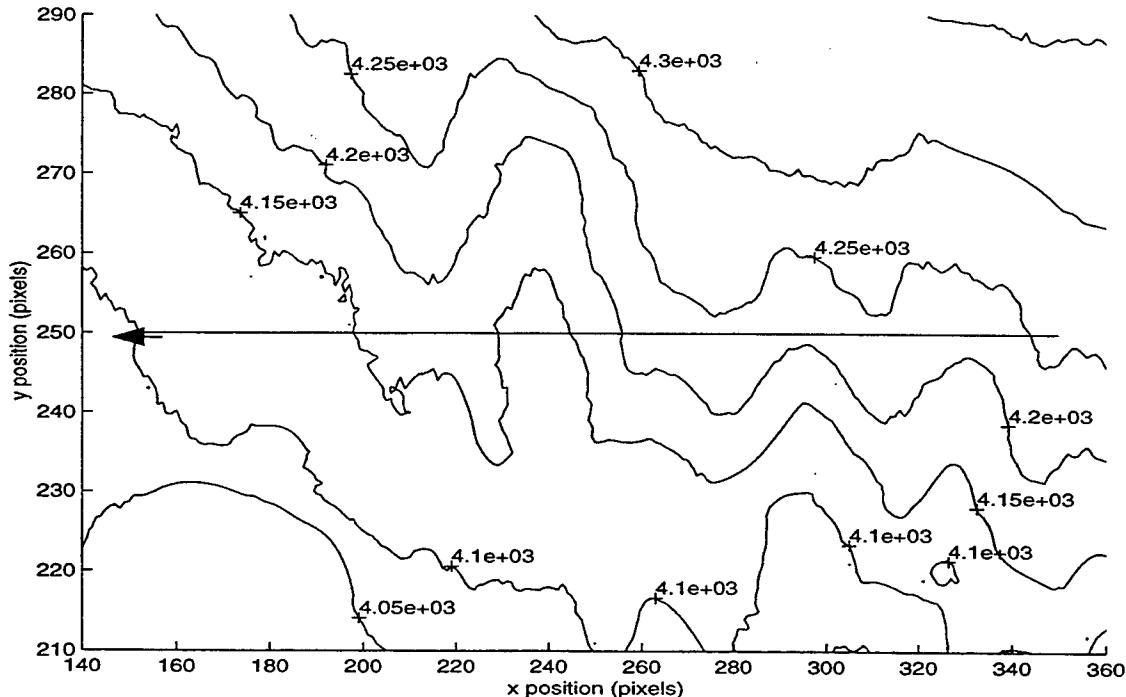
Figure 5.19. Region 5 with Trackline 5.

neighborhood of Trackline 5: Figures 5.20 and 5.21 show the values of R_{xx} and R_{yy}. For the entire trackline, both R_{xx} and R_{yy} stay fairly small. In fact R_{yy} stays below one for the

116
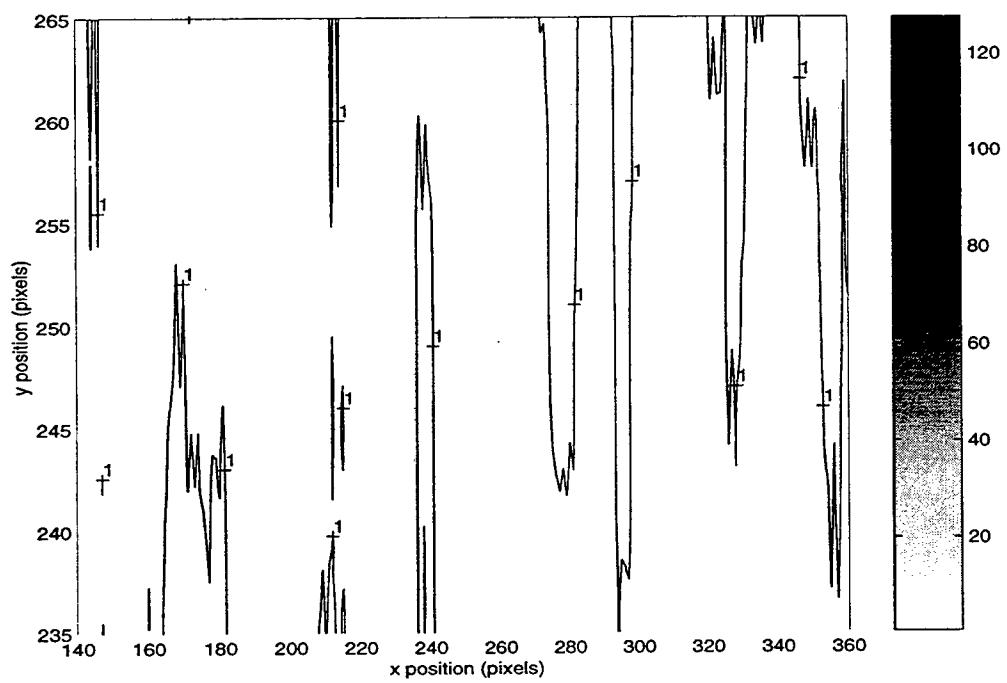
Figure 5.20. R_{xx} for Trackline 5.
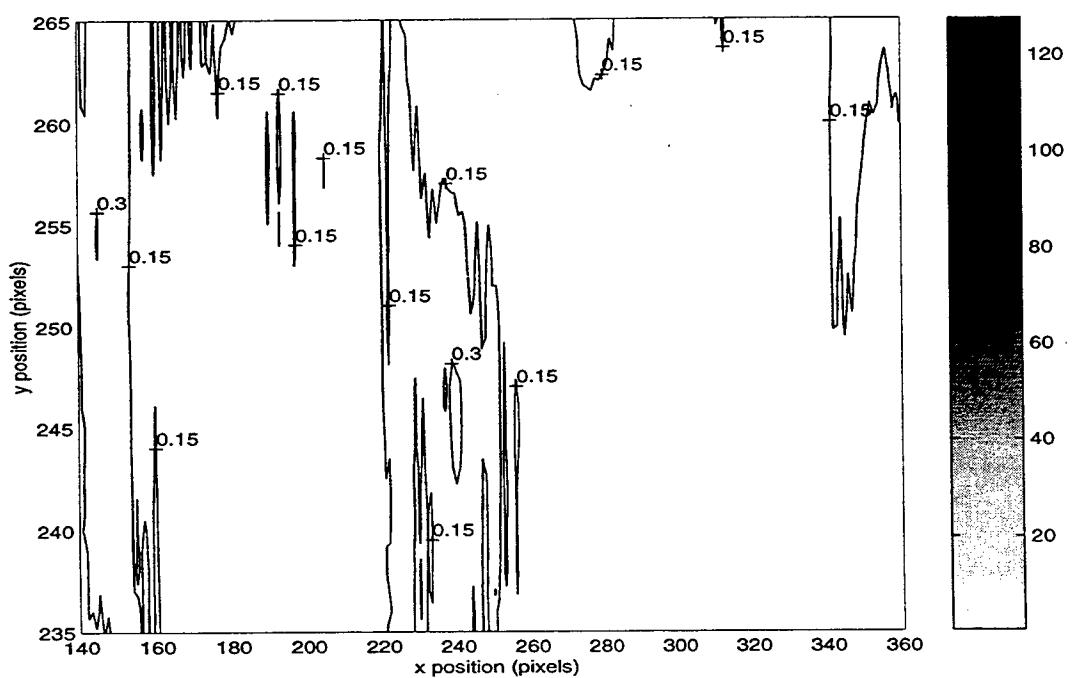
Figure 5.21. R_{yy} for Trackline 5.

entire region.

Figures 5.22 and 5.23 show the true and navigated tracks and the associated

covariances for Trackline 5. The navigated track never differs from the true track by more

than one pixel, and the covariance is larger than one only once. Tracking in this region is

extremely good.
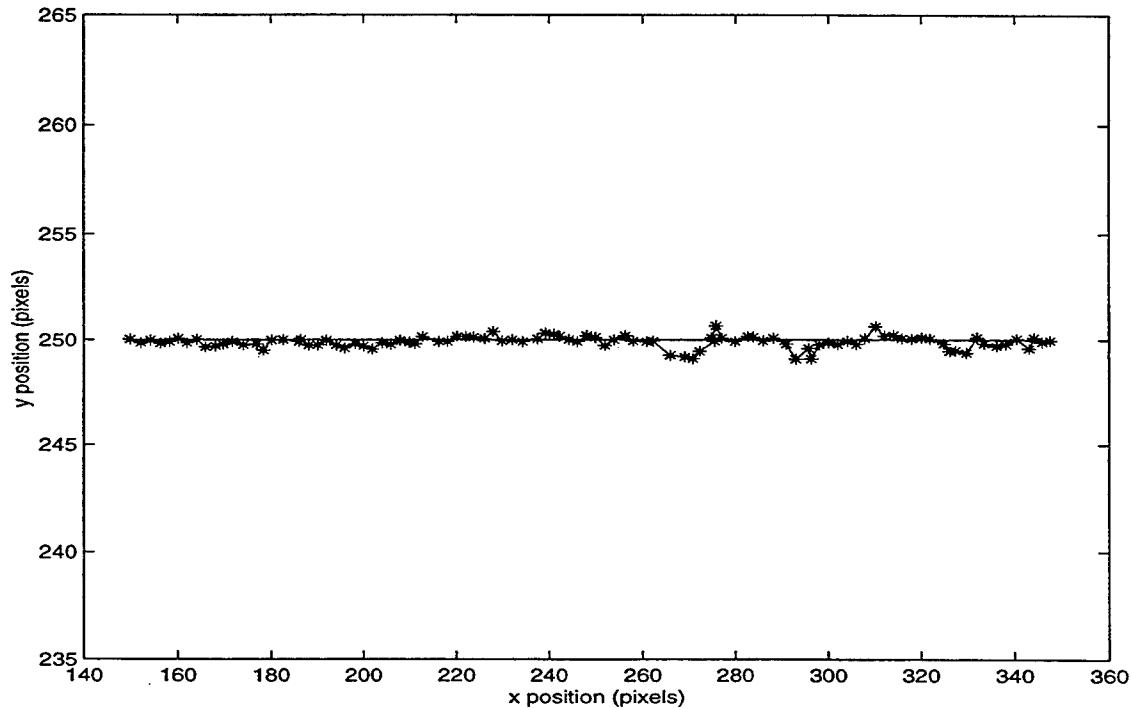
Figure 5.22. True (-) and Navigated (*) Trackline 5.

5.3 Quantitative Analysis

This section presents a quantitative analysis in true dimensions (meters) of the five

tracklines given in Section 5.2.

Figures 5.24 show the overall distance error between the true and navigated tracks,

along with the errors in x and y for Trackline 1 (see Figures 5.1 and 5.4). The average x, y,
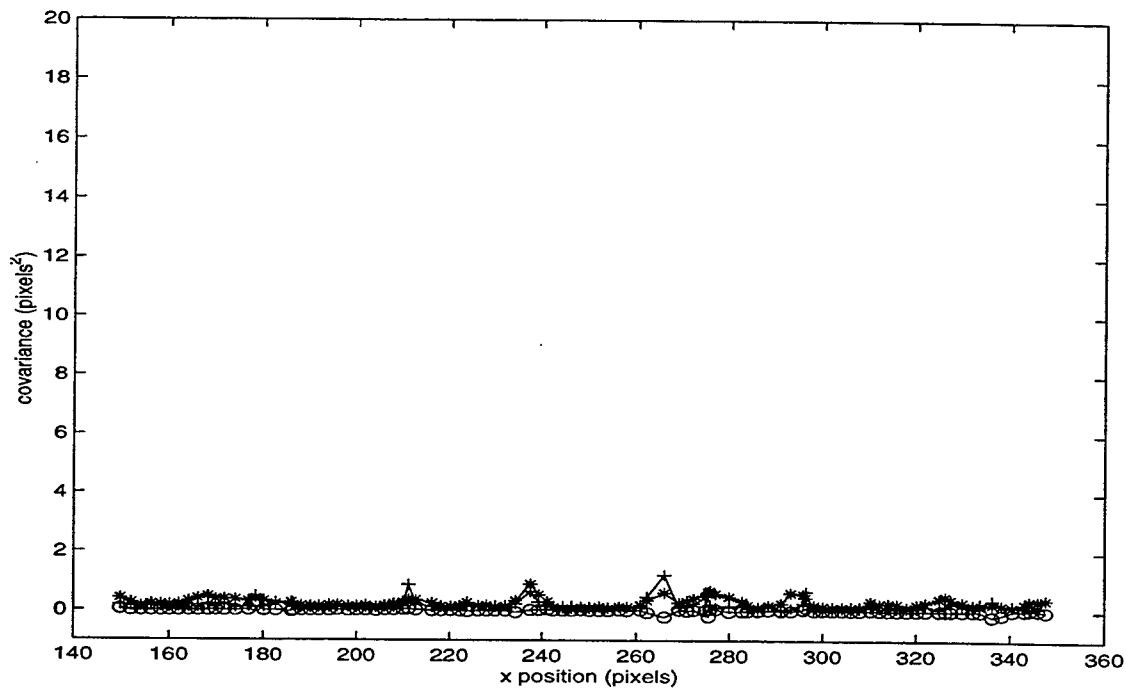
Figure 5.23. Covariances for Trackline 5.
xx(*), yy(+), xy(o).

and overall errors are 5.67 m, 4.14. m, and 7.82 m, respectively, for the full trackline, but

the plots of Figures 5.24 clearly show three separate sections of navigation performance.

These three sections correspond to the three types of terrain this trackline covers. The first

section contains time steps 1 through 45. Average errors are 4.95 m and 7.65 m for the x

and y directions, and 9.70 m for the overall average error. The second section is for time

steps 46 through 80 and average errors are 0.64 m, 0.70 m, and 1.05 m for x, y, and overall

error, respectively. The final section is for time steps 81 through 150, and the average

errors are 8.65 m and 3.61 m for x and y, with an overall average error of 9.99 m.

The covariance plots of Section 5.2 contain all necessary covariance information and

are not reproduced here. The covariance values, given in these figures in units of pixel^2,

should be multiplied by 25 to obtain true covariance values in m^2. For Trackline 1, average

covariances for x, y, and xy are 39.93 m^2, 35.40 m^2, and -0.02 m^2, respectively (see Figure
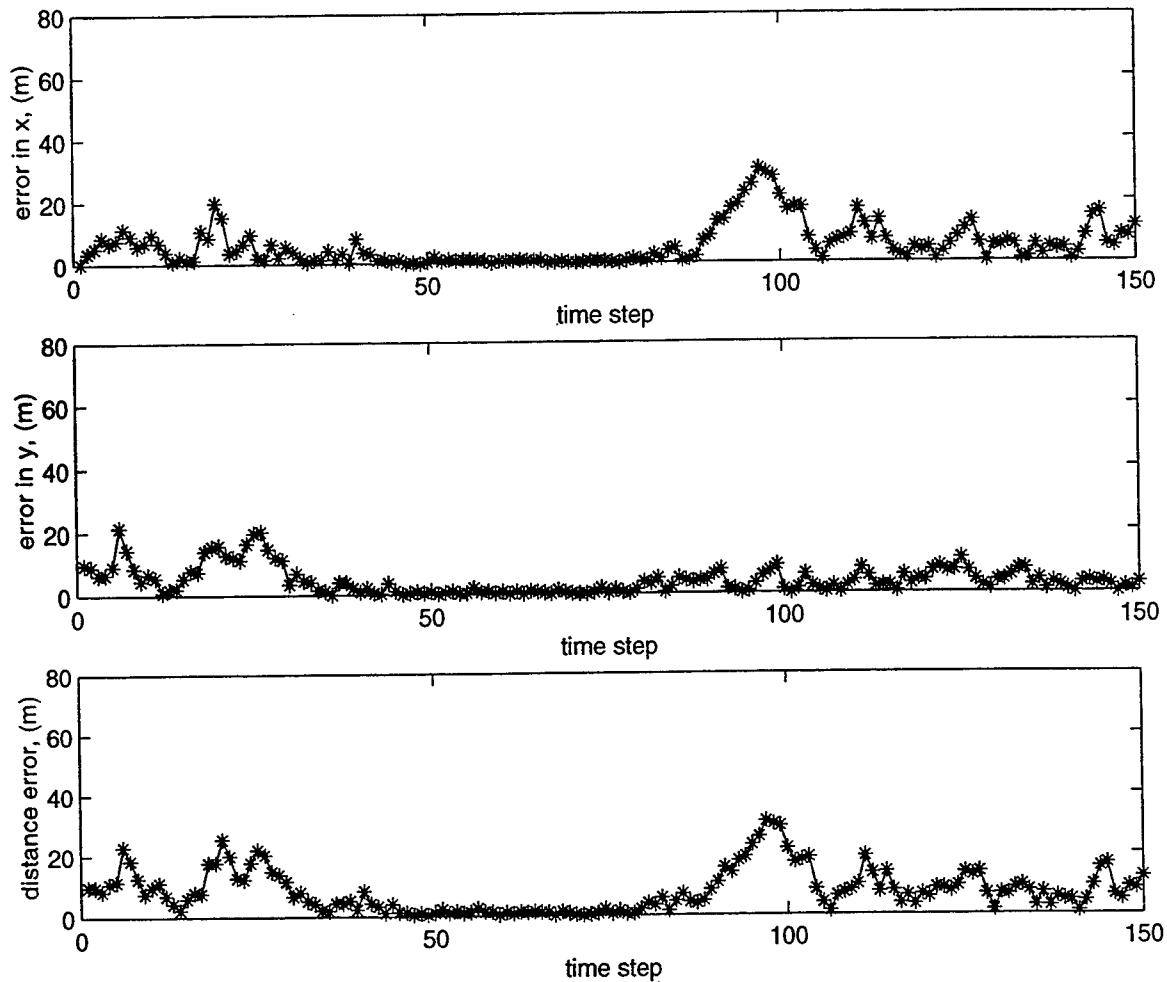
Figure 5.24. *x, y,* and Overall Errors for Trackline 1.

5.5). Segmenting the covariances as above, for Section 1, the average values for x, y, and

xy are 67.00 m^2, 78.50 m^2, and 1.25 m^2; for Section 2, 3.96 m^2, 3.15 m^2, and 0.00 m^2; and

for Section 3, 40.47 m^2, 23.88 m^2, and -0.85 m^2. However, the covariance information is

more useful when compared to the error at each time step.

Figure 5.25 shows the track error normalized over the covariances of Figure 5.5. The

overall distance error is compared to the state covariance matrix at that time step to obtain

the value of T, which indicates the size of the uncertainty ellipsoid that corresponds to the

overall distance error vector. For example, when $T=1$, an ellipsoid is drawn at 1σ, and the
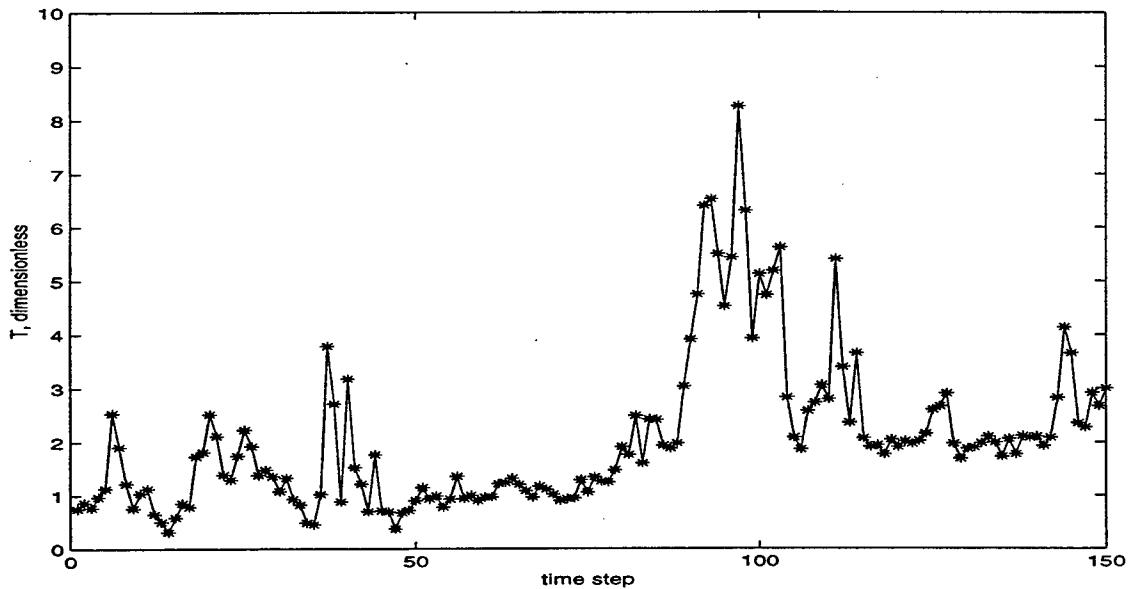
Figure 5.25. Track Error Normalized over Covariance (*T*) for Trackline 1.

endpoint of the difference vector lies on the surface of this ellipsoid. When *T*=2, an

ellipsoid is drawn at 2σ, and the endpoint of the difference vector lies on the surface of

this ellipsoid. So, the value of *T* is the number of sigmas needed to define the ellipsoid

whose surface contains the endpoint of the difference vector. At each time step, the actual

deviation values are different, but for this figure the information is normalized so different

time steps can be compared directly.

At the start of the track, the overall distance error is large (see Figure 5.24), but so is

the covariance (see Figure 5.5), and the performance of the system is fairly good, i.e., *T* is

small. In the second section, the errors are smaller, but so are the covariance values, and

the performance is still good. However, when the vehicle moves into the third section of

the terrain, the error values become large again, but the covariance has not yet grown

sufficiently, so the normalized value is large (up to 9), indicating poor performance. This

value does reduce as the covariance grows to better represent the accuracy expected.
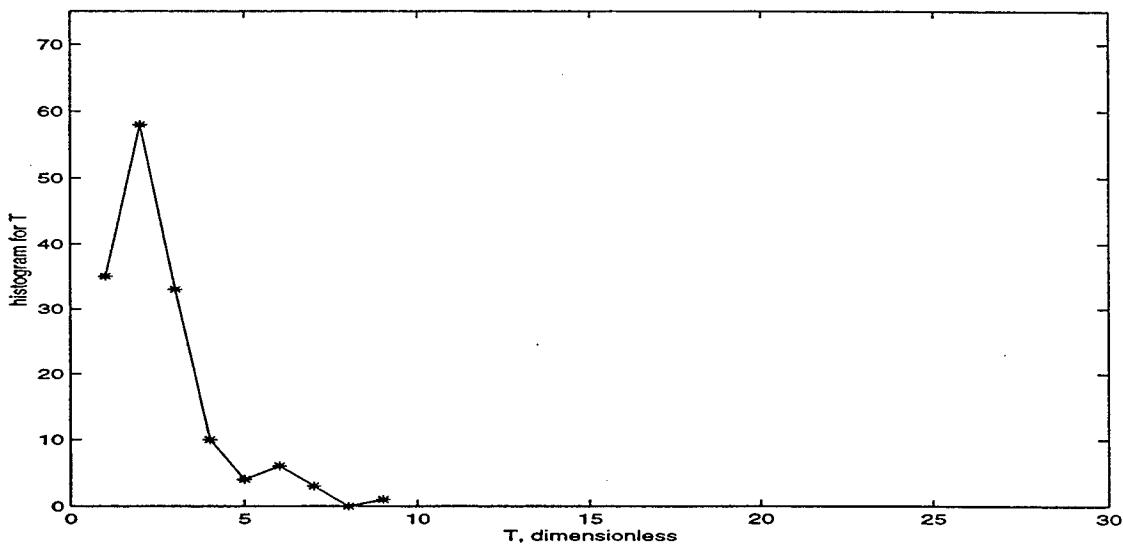
Figure 5.26. Histogram for T, Trackline 1.

Figure 5.26 shows a histogram of T for all of Trackline 1. The histogram indicates that for most of the trackline T is small. In fact, for 84% of the track, $T<3$, or the error is within 3 standard deviations of the calculated uncertainty. These results show very good performance but also indicate that the state covariance values are slightly smaller than they should be. For a perfect system, T should follow the gaussian distribution where 99% of the track should have $T<3$.

Figures 5.27 show the overall distance error between the true and navigated tracks along with the errors in x and y for Trackline 2 (see Figures 5.1 and 5.7). Here, the average x, y, and overall errors are 5.36 m, 4.36 m, and 7.82 m, respectively. The overall average covariances for x, y, and xy, respectively, are 42.74 m^2, 40.74 m^2, and -2.53 m^2 (see Figure 5.8). The plots of Figures 5.27 also show three separate regions of navigation performance, as they should, since Trackline 2 is the reverse of Trackline 1. Here, the regions are separated as time steps 1 through 70, 71 through 105, and 106 through 150. For Section 1, average errors are 8.94 m, 4.14. m, and 10.69 m for x, y, and overall,

122
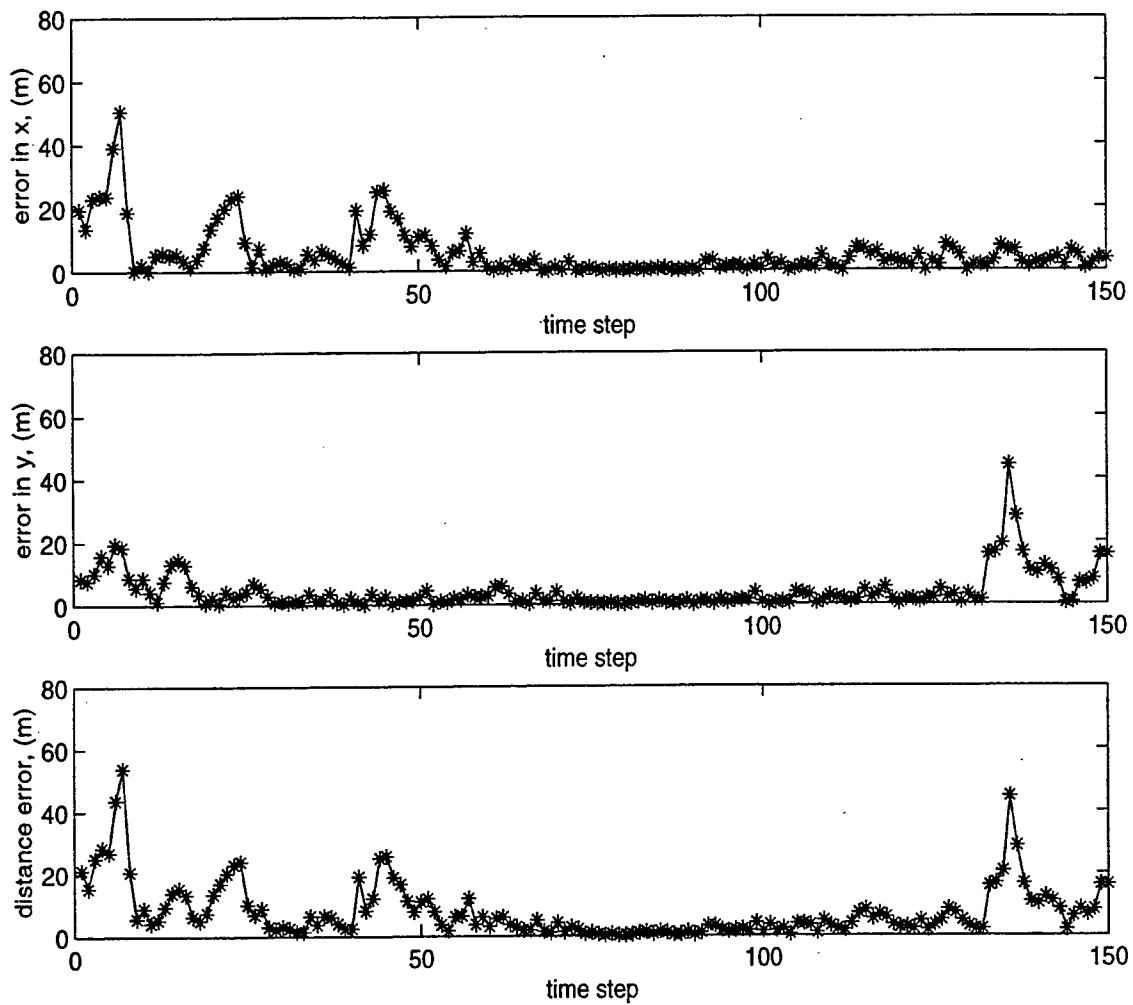
Figure 5.27. *x*, *y*, and Overall Errors for Trackline 2.

respectively. This corresponds to Section 3 of Trackline 1, which has comparable error

values. The average covariances for *x*, *y*, and *xy*, respectively, are 47.88 m^2, 43.81 m^2, and

-6.18 m^2. For Section 2, the average errors are 0.86 m, 0.91 m, and 1.43 m for *x*, *y*, and

overall, respectively, and again the corresponding error values for Trackline 1 are

approximately the same. The average covariances for *x*, *y*, and *xy*, respectively, are 3.80

m^2, 3.07 m^2, and 0.02 m^2. Section 3 has average errors of 3.30 m, 6.78 m, and 8.31 m for

x, *y*, and overall, respectively, and in this case the *y* error, and consequently the overall

error, differs slightly from those obtained for Section 1 of Trackline 1. Here, the average

covariances for x, y, and xy are 65.03 m^2, 1.17 m^2, and 65.28 m^2.

Figure 5.28 shows Trackline-2 error normalized over the covariances of Figure 5.8. Here, the navigation performance is suboptimal in the beginning as the values for T are larger than desired. This is consistent with Section 3 of Trackline 1 (see Figure 5.25). The largest values of T for this trackline occur at approximately the same locations in the map as the largest values of T for Trackline 1. Here, however, the largest value is 23 compared
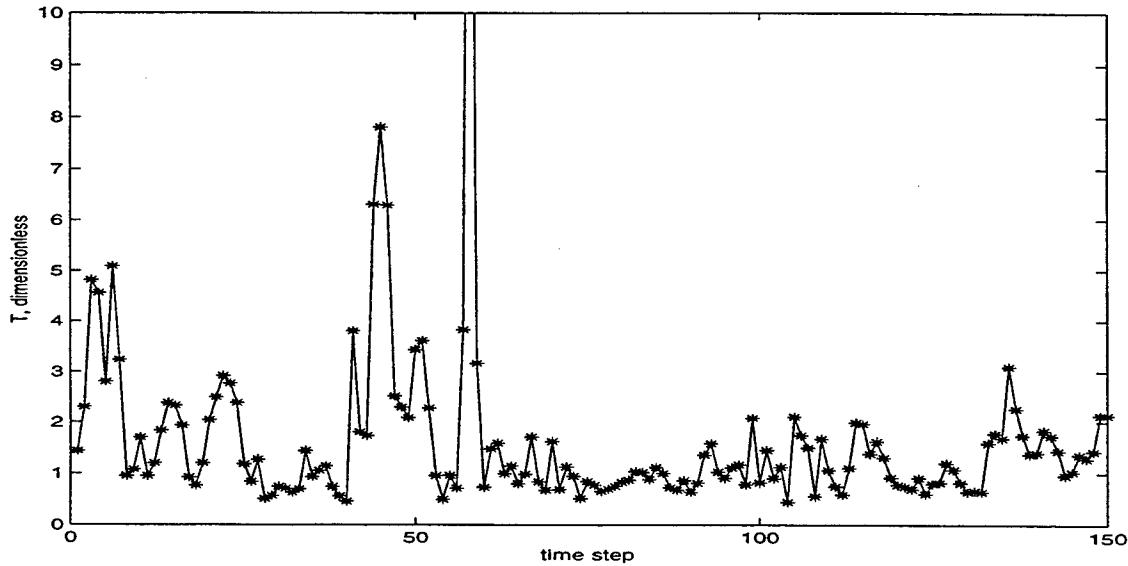
Figure 5.28. Track Error Normalized over Covariance (T) for Trackline 1.

to 9 of the other trackline. The area where this occurs is the transition area between the steep and the gradually sloping terrains. The second section of Figure 5.28 represents the ridge area of the terrain, and the values of T indicate good performance, similar to Section 2 of Trackline 1. The final section shows a slight degradation in performance, but overall the system operates quite successfully.

Figure 5.29 shows a histogram of T for all of Trackline 2. The histogram shows that for most of this trackline, as for the previous one, T remains small. In fact, for 91% of the
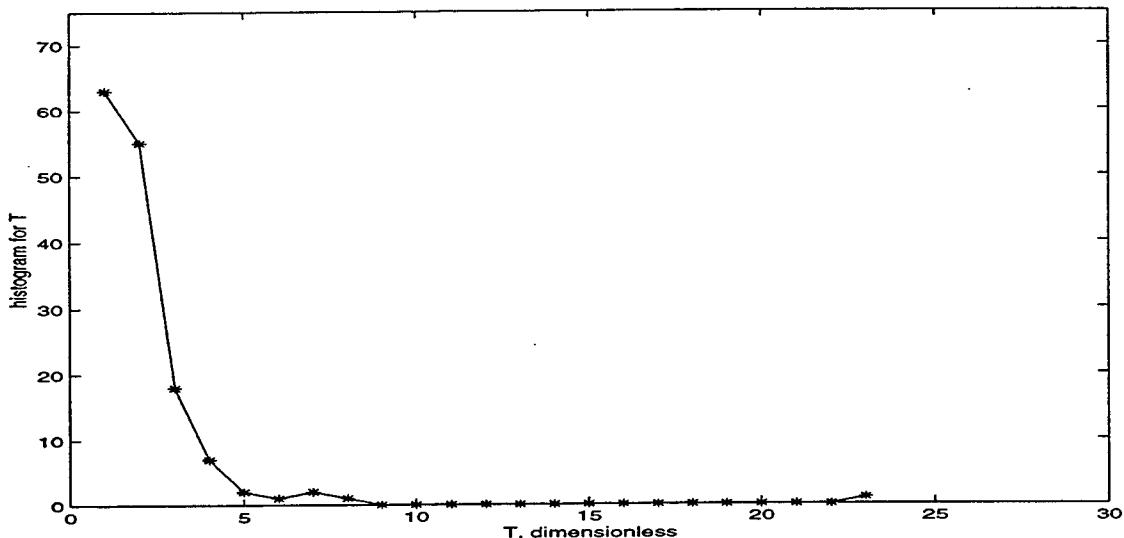
Figure 5.29. Histogram for T, Trackline 2.

track, $T<3$, which indicates better performance than Trackline 1. The only exception is the single outlier at $T=23$, as discussed above. However, as in Trackline 1, the state covariance values are still slightly smaller than they should be for a perfect Gaussian system.

Figures 5.30 show the overall distance error between the true and navigated tracks, along with the errors in x and y for Trackline 3 (see Figures 5.9 and 5.12). The average x, y, and overall errors are 3.50 m, 3.23. m, and 5.12 m, respectively. The majority of this trackline is for the steep ridge terrain (all except the very beginning), and it is not necessary to segment it as for Tracklines 1 and 2. The average covariances for this track are 17.86 m^2, 14.27 m^2, and -3.44 m^2, for x, y, and xy, respectively (see Figure 5.13)

Figure 5.31 displays the normalized track error for Trackline 3, and Figure 5.32 the corresponding histogram. The histogram is very similar to that of Trackline 2 (Figure 5.29) in that it has isolated points at a high T value (28), and 90% of the track has $T<3$. The two points for which $T=28$ are located in different places. The first point occurs at the transition into the steep terrain, and the second occurs when the covariances are extremely

125
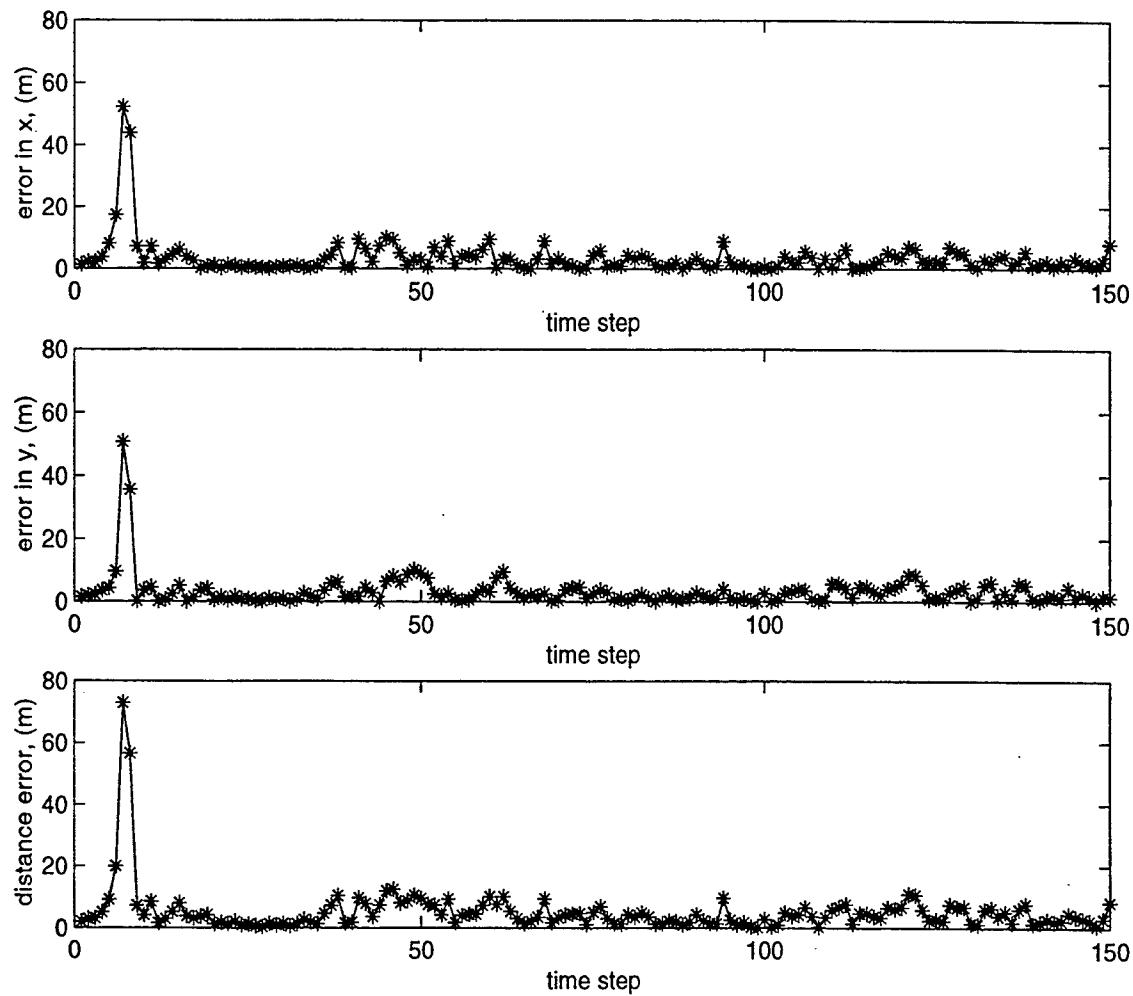
Figure 5.30. x, y, and Overall Errors for Trackline 3.

small. Isolated points like these, although undesirable, do not result in overall poor

performance of the system. Again, there is very good performance for this track.

Figures 5.33 show the overall distance error between the true and navigated tracks,

along with errors in x and y, for Trackline 4 (see Figures 5.14 and 5.17). The average x, y,

and overall errors are 15.42 m, 4.47 m, and 16.62 m, respectively, for the full trackline.

The average covariances are 49.85 m^2, 40.92 m^2, and -3.65 m^2, for x, y, and xy,

respectively (see Figure 5.18). This trackline can be divided into two sections, one

covering the gradually sloping terrain, and the other covering the ridge. The first section
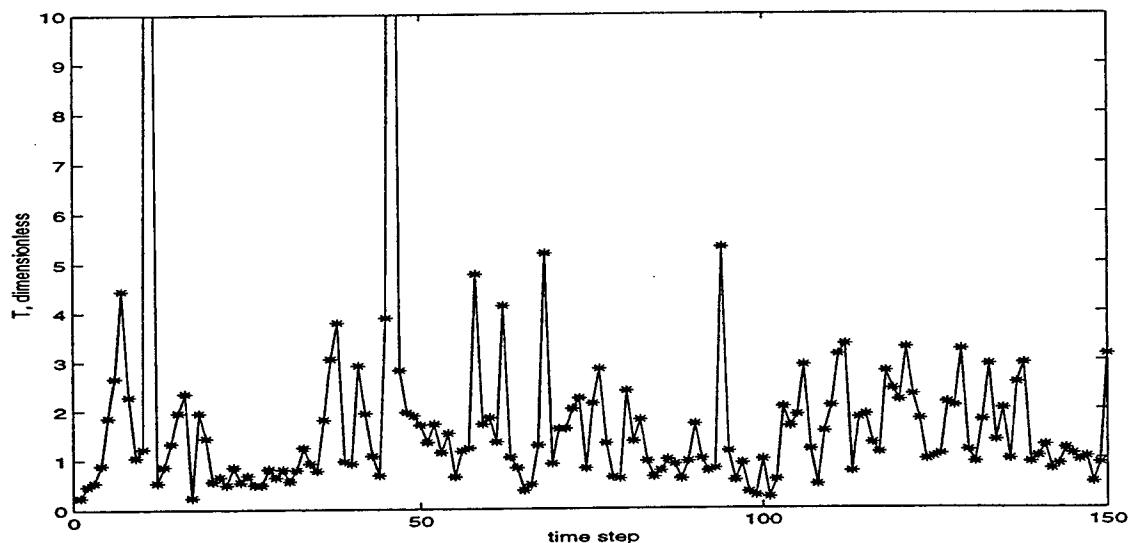
Figure 5.31. Track Error Normalized over Covariance (*T*) for Trackline 3.
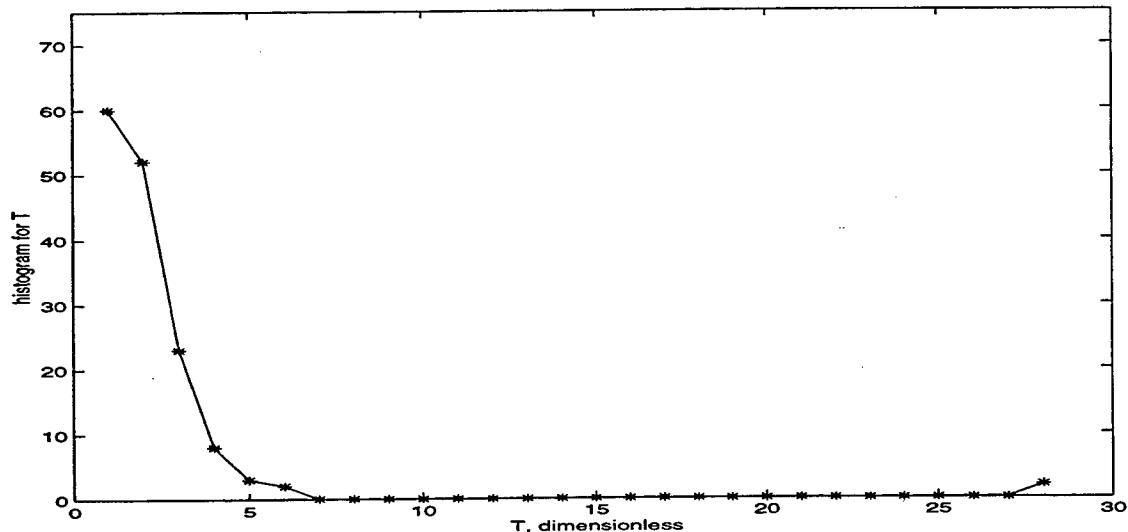
Figure 5.32. Histogram for *T*, Trackline 3.

contains time steps 1 through 65, and the average *x, y,* and overall errors are 22.17 m, 5.40

m, and 23.28 m, respectively. The average covariances are 69.56 m², 55.47 m², and -5.26

m², for *x, y,* and *xy,* respectively. The second section contains time steps 66 through 100,

and the average *x, y,* and overall errors are 2.88 m, 2.74 m, and 4.26 m. Here, the average

covariances are 13.26 m², 13.90 m², and -0.65 m², for *x, y,* and *xy,* respectively. Although

this trackline covers general areas of the terrain that are already visited in the first three
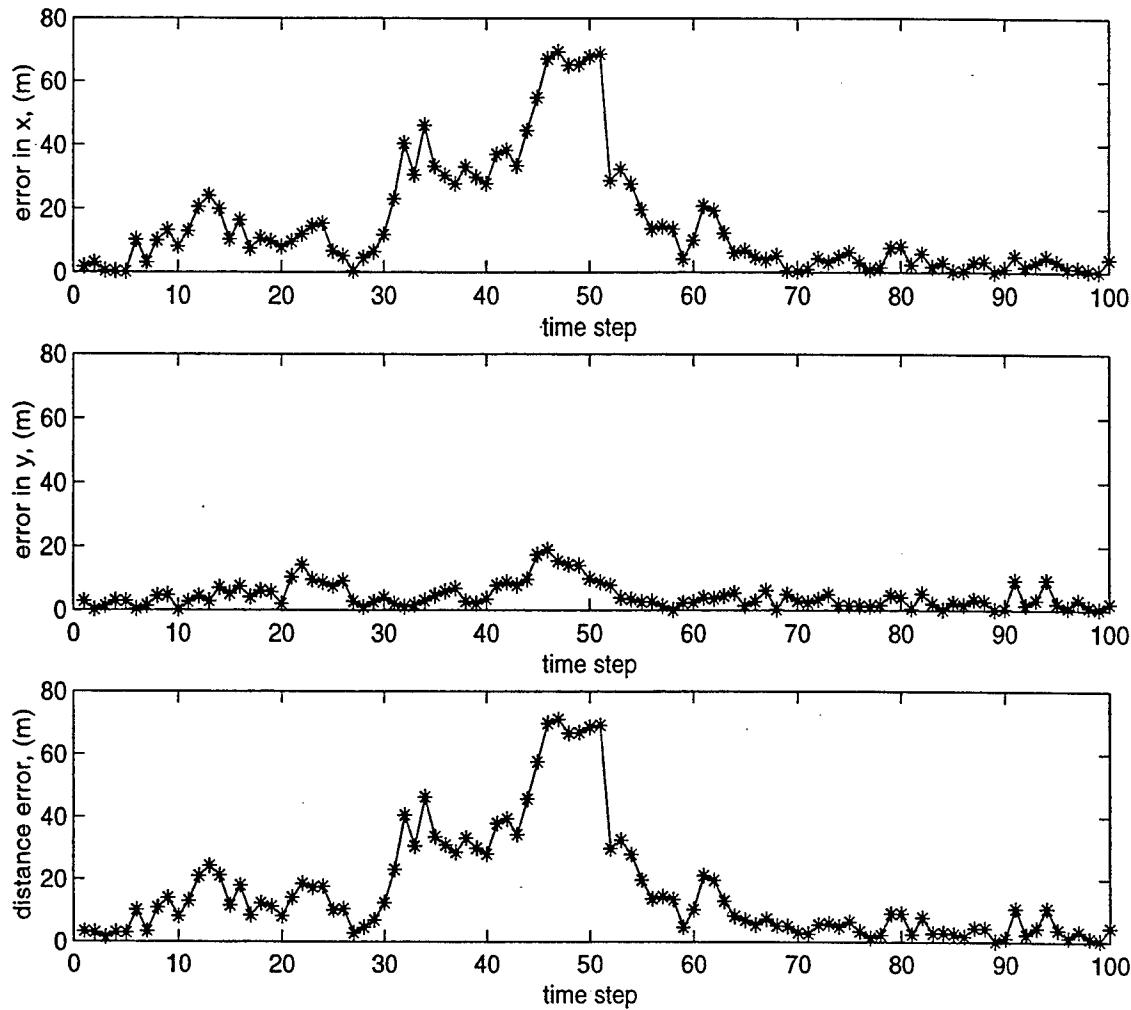
Figure 5.33. *x, y,* and Overall Errors for Trackline 4.

tracklines, in this case the profiles are oriented perpendicular to the profile orientation of

the previous tracks, so very different results are possible.

Figure 5.34, shows Trackline-4 error normalized over the covariances of Figure 5.18.

Performance for this trackline is not as good as that of the three previous tracks.

Undesirably large values of T are sustained for over fifteen time steps before T reaches its

maximum value of 11 and then decreases. The reduction occurs starting at time step 50,

and T stays small for the remainder of the track, except for a few isolated spikes that reach

almost to 4. The first half of this trackline, where the performance is marginal, occurs in
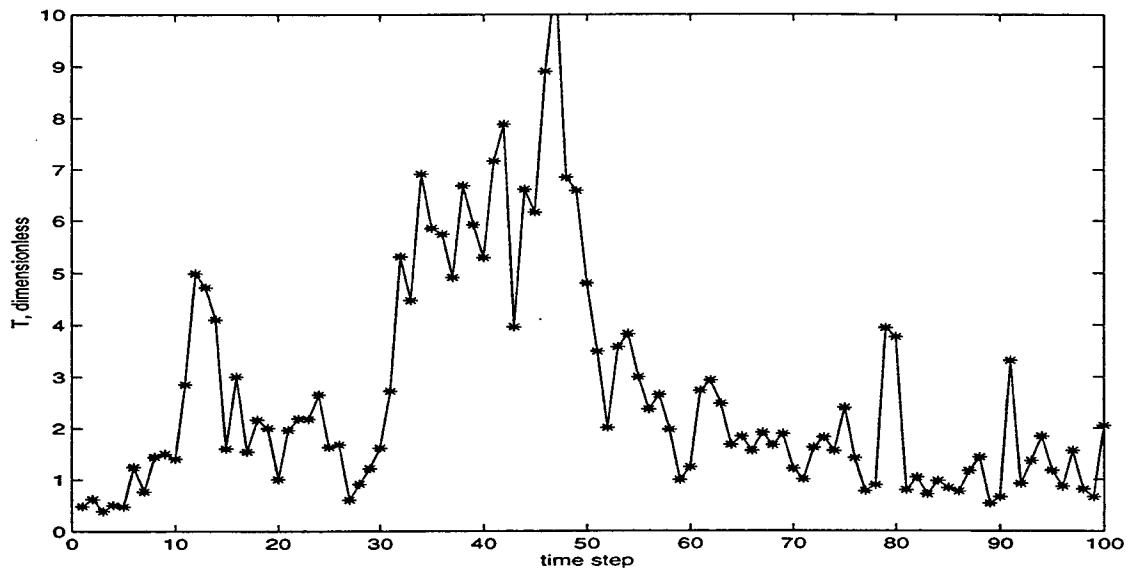
Figure 5.34. Track Error Normalized over Covariance (*T*) for Trackline 4.

the terrain that is much less variable than the terrain for the second half of the track (see

Figure 5.14). The transition into the ridge occurs roughly in the middle of the track,

approximately at time step 50, when the *T* values begin to decrease.

The sub-optimal, yet acceptable, performance across this track is evident in the

histogram of *T* as shown in Figure 5.35. Although there are no outliers, only 71% of the
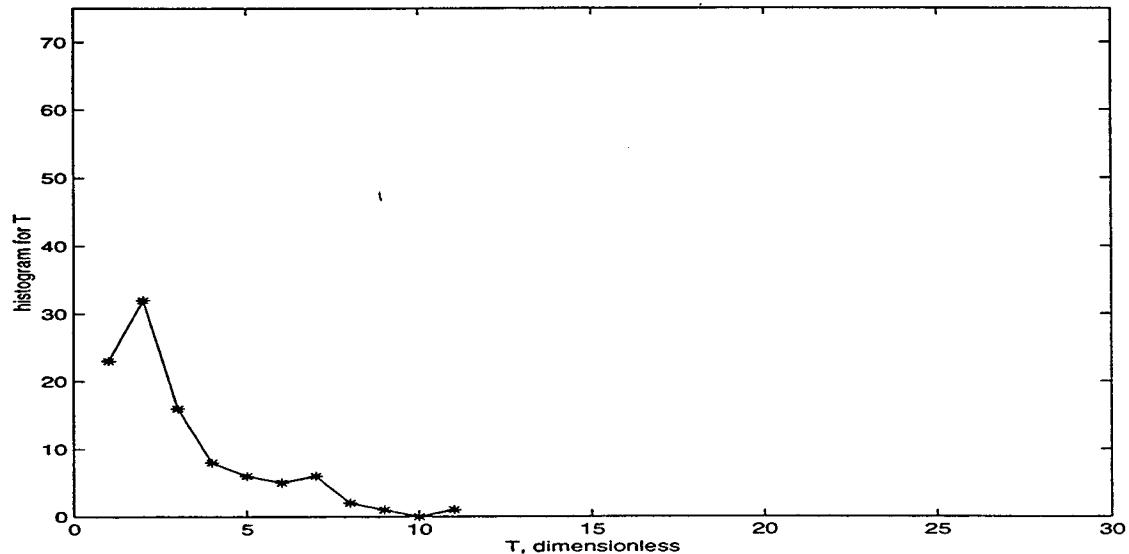
Figure 5.35. Histogram for *T*, Trackline 4.

129

track has $T<3$. As can be seen in the figure, the covariances at the beginning of the track are smaller than they should be.

Figures 5.36 show the overall distance error between the true and navigated tracks
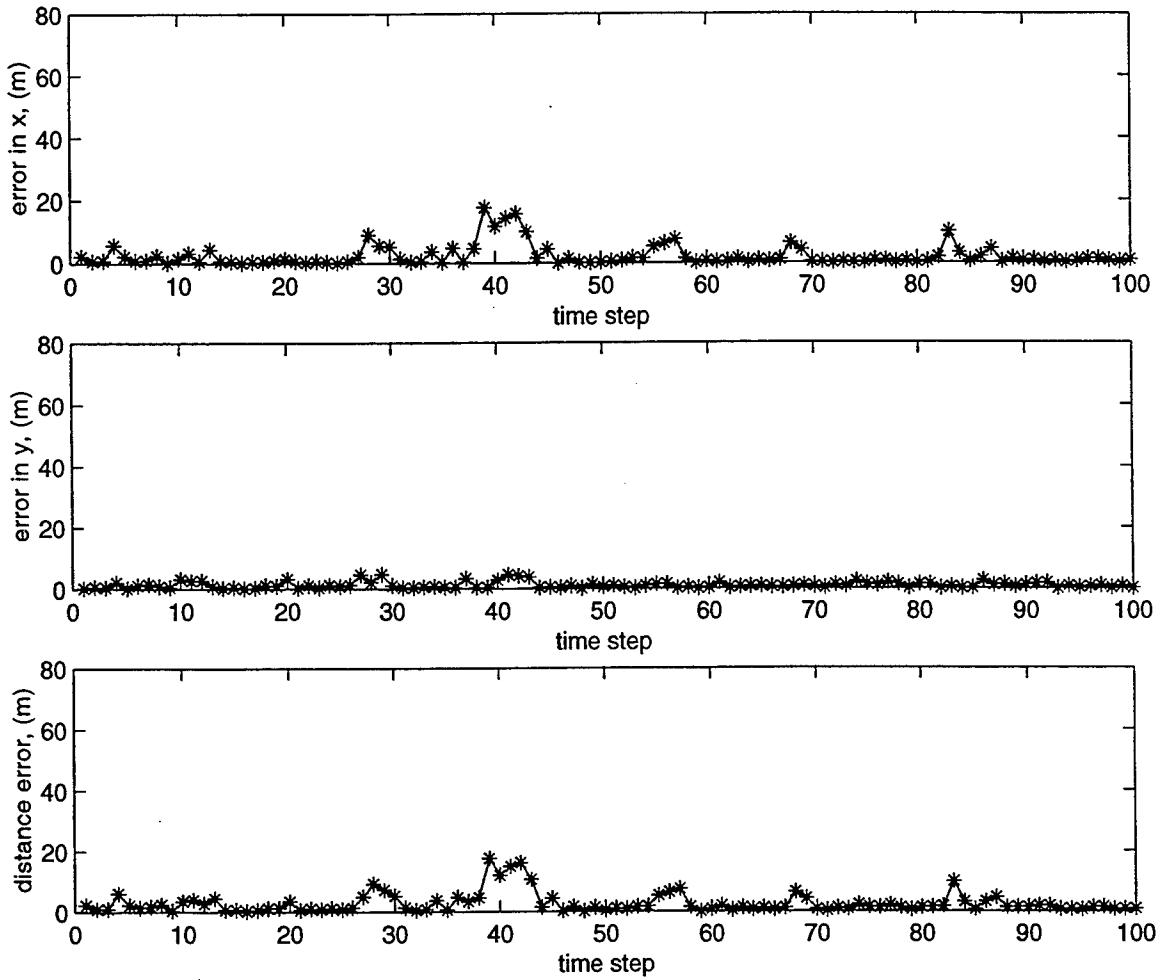
Figure 5.36. x, y, and Overall Errors for Trackline 5.

along with the errors in x and y for Trackline 5. This entire trackline is within the steep ridge, so it is not segmented. The average x, y, and overall errors are 2.24 m, 1.08 m, and 2.75 m, respectively. The average covariances are 6.75 m^2, 4.39 m^2, and -0.02 m^2, for x, y, and xy, respectively. As a complete trackline, Trackline 5 has the smallest navigational errors; only the second sections of Tracklines 1 and 2 have smaller errors. Also, the

covariances are the smallest for this complete trackline. Only the second sections of Tracklines 1 and 2 have smaller covariances, and Section 2 of Trackline 4 has covariances only slightly larger

Figure 5.37 shows the normalized track error for Trackline 5 and, Figure 5.38 the corresponding histogram. The two places where T remains larger than 3 for consecutive time steps correspond to the places where the x error is large.The maximum T value of 11
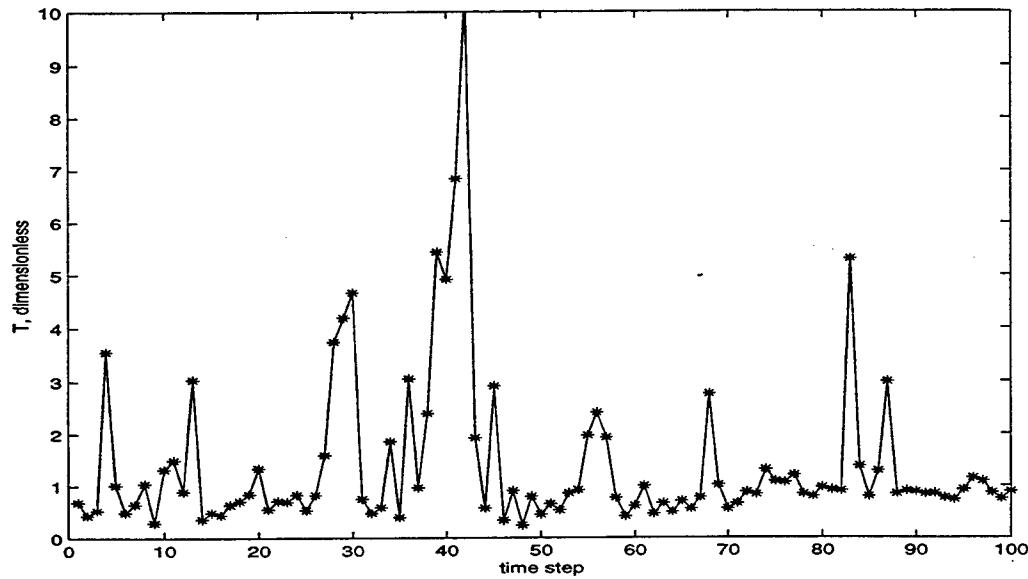
Figure 5.37. Track Error Normalized over Covariance (T) for Trackline 5.

occurs only once, and for 89% of the track $T<3$, which gives performance almost as good as Tracklines 2 and 3 (91% and 90%, respectively). Also, 84% of Trackline 5 has $T<2$, which shows significantly better performance that Tracklines 1 and 4, where $T<3$ for 84% and 71%, respectively.
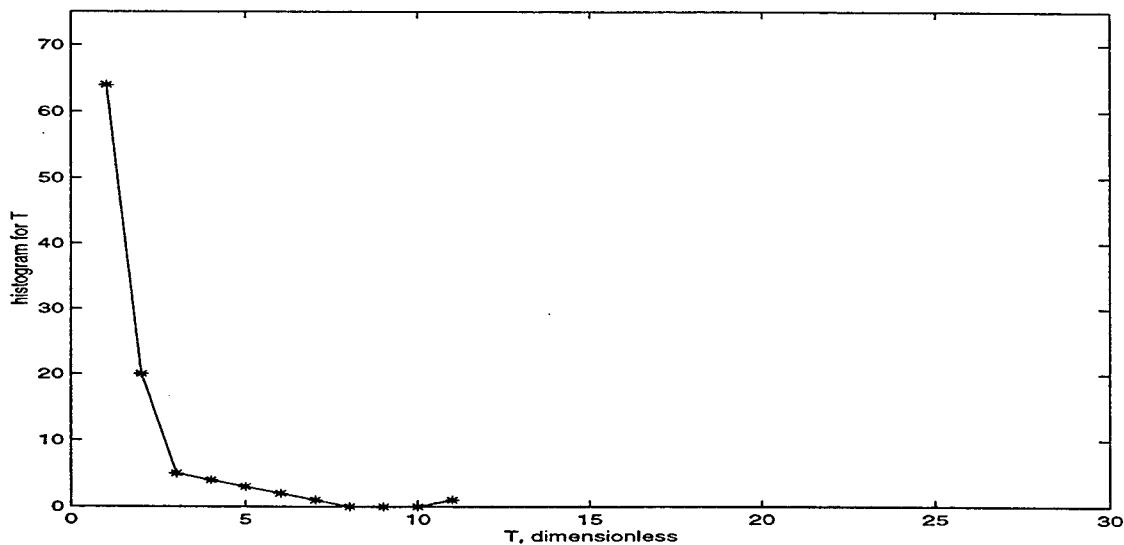
Figure 5.38. Histogram for T, Trackline 5.

5.4 Summary

Five tracklines across different types of terrain in deep ocean bathymetry were presented in this chapter, for which the results are summarized in Table 5.2 The tracks crossing the steep ridge with highly variable terrain show the smallest navigation errors (Trackline 5 and Sections 2 of Tracklines 1, 2, and 4). In these regions the errors are smaller than the pixel spacing of 5 m. In all these regions, the covariances are also smaller than the pixel spacing, i.e., less than 25 m^2. The worst performances are, as expected, in areas of less variable terrain, with overall errors up to 23 m and covariances up to 78 m^2. However, no navigation track ever diverged from the true track beyond the point of recovery. In this sense, all tracklines were very successful.

Table 5.2 Summary of Trackline Performance Data.

Trackline	x error, m	y error, m	overall error, m	x covariance m^2	y covariance m^2	xy covariance m^2	% T < 3
1	5.67	4.14	7.82	39.93	35.40	-0.02	84
2	5.36	4.18	7.82	42.74	40.74	-2.53	91
3	3.50	3.23	5.12	17.86	14.27	-3.44	90
4	15.42	4.47	16.62	49.85	40.92	-3.65	71
5	2.24	1.08	2.75	6.75	4.39	-0.02	89
1-1	4.95	7.65	9.70	67.00	78.50	1.25	
1-2	0.64	0.70	1.05	3.96	3.15	0.00	
1-3	8.65	3.61	9.99	40.47	23.88	-0.85	
2-1	8.94	4.14	10.69	47.88	43.81	-6.18	
2-2	0.86	0.91	1.43	3.80	3.07	0.02	
2-3	3.30	6.78	8.31	65.03	1.17	65.28	
4-1	22.17	5.40	23.28	69.56	55.47	-5.26	
4-2	2.88	2.74	4.26	13.26	13.90	-0.65	

Chapter 6

Conclusions

In this thesis, I demonstrate the success of a new terrain-relative navigation algorithm for autonomous underwater vehicles. The approach does not require the deployment of acoustic beacons or other navigation aids. This means there is no need for servicing and maintaining any equipment other than the vehicle itself. Also, no time or effort is spent surveying by support vessel to establish the beacon locations or to define a local reference frame. Instead, terrain-relative navigation depends on a supplied digital bathymetric map and the ability of the vehicle to image the seafloor. These images are matched to the map in the local neighborhood and ranked according to the mean absolute difference. The algorithm functions independent of the pixel size of the supplied map, so even maps with coarse resolution can be used successfully.

An intelligent, adaptive search, using a coarse-to-fine algorithm with a modified beam search, reduces the computational requirements at each time step. This should enable real-time operation in most environments. Actual computational requirements will vary for each time step as a function of the size of the search range, which dictates the number of coarse maps, and the number of matches tracked through the scale space. Validated measurements are weighted probabilistically as a function of both the match quality and the innovation using a probabilistic data association filter with amplitude information (PDAFAI). This means that all good matches are considered with some probability of being the correct match.

Five simulated tracklines were run across varying terrain in a deep-ocean map. As expected, results show more accurate navigation in areas with greater bathymetric variability and less accurate navigation in flatter areas with more gentle terrain contours. Terrains with minimal bathymetric variability, such as the abyssal plains, may not yield profiles with sufficient uniqueness to support accurate navigation with this approach. For most places in the tracklines presented, the uncertainties assigned to the navigation positions reflect the ability of the system to follow the true track. In no case did the system diverge beyond the point of recovery.

6.1 Assumptions

The assumptions for this research fall into two categories: operational and conceptual. The two operational assumptions are, as stated above, that there is a supplied digital bathymetric map and that the vehicle has the ability to image the seafloor. To have a supplied map implies that the area of interest has been visited previously. Thus, this terrain-relative navigation algorithm is not applicable when the terrain is unknown. In the simulations of Chapter 5, it is also assumed that the seafloor image is a 1D string of digital bathymetric values with uncertainty. (The algorithm is easily adapted to handle 2D images.) This means that regardless of the sensor used, the ability to convert the sensed information into a digital bathymetric string is assumed.

One primary conceptual assumption is that all uncertainties are zero-mean Gaussian. Each map point and each profile point has an associated uncertainty value. For the map, uncertainty values are assumed based on both the sensor used to obtain the information and the map-making technique. The profile uncertainties due to the sensor are also

assumed constant. This is not likely to be true. One would expect the data at the ends of each profile to maintain higher uncertainty because the slant ranges are usually larger. One would also expect the uncertainty to change as a function of the altitude of the vehicle.

The zero-mean Gaussian assumption is fundamental to the PDAFAI. As stated in Chapter 3, with this methodology, the past is summarized by assuming the state is normally distributed according to the latest estimate and covariance matrix. Further, each new estimate with uncertainty is found using only the measurements at that time step.

However, the most critical assumption of this research is that it is better to have an estimate with larger uncertainty than to choose the wrong measurement and have the vehicle truly believe it is somewhere that it is not. In other words, *it is better to accept less accuracy than it is to risk being wrong*. This assumption is the motivation behind using multiple measurements and the PDAFAI, which is a major contribution of this research.

6.2 Contributions

This thesis provides several major contributions to the field of navigation for underwater vehicles. Although it is not the first work to investigate the use of underwater terrain for map-based navigation, it is the first to accept multiple matches from the map-matching algorithm. This is the cornerstone of this research and its greatest advantage. All location matches in the vicinity of the expected location are considered with some probability of being the true location. The weighting on each is derived from the "goodness" of the match as well as its proximity to the expected value. Also, this research is the first application of probabilistic data association techniques to calculating new position updates using measurements derived from a map-matching algorithm.

A second contribution involves the computational requirements of map-matching in large search areas. This thesis shows that the techniques of coarse-to-fine matching and beam searching can be applied to map-matching navigation with only minor modifications. Using these techniques, computational requirements for each navigation update can be reduced significantly. Application of these techniques can affect overall algorithm performance, and the potential impacts are evaluated.

Another contribution is in the area of handling uncertainties. Other navigation systems track and explicitly represent uncertainties on the vehicle location and sensed information. However, this research additionally allows the uncertainty of the supplied map to impact the system performance. No other bathymetric navigation system accounts for map uncertainties. Because this is integrated into the algorithm, navigation is possible even with highly uncertain maps or with maps with a range of uncertainties.

This research also provides a method for the type of terrain to affect navigation accuracy. This occurs through the measurement noise covariance matrix. Highly variable terrain produces small measurement covariances and should enable highly accurate navigation, while flatter terrain produces higher measurement covariances and navigation accuracy should be reduced. At this time, quantifying the navigation accuracy based solely on the terrain is difficult because other parameters also affect accuracy. However, such a study could result in a major contribution to this field of research.

6.3 Further Directions

The research of this thesis can be expanded in several ways. First and foremost, the system could be implemented on one of several existing AUVs with the necessary sensors

on board. The navigation could then be tested in areas where bathymetric maps of varying resolution and accuracy are already available. To achieve this, a more detailed model of the vehicle system dynamics is necessary. Also required is the automated transformation of raw sensor data to a regularly spaced digital profile. These subsystems have already been developed by other researchers.

One option for expanding the capabilities of the system is to add the backscatter strength to the PDAFAI. In areas where the bathymetric variation is not sufficient to support the desired navigational accuracy, the amplitude of the sonar signal could provide additional information. For example, a flat rocky bottom will produce a much different signal than flat and muddy or flat with heavy aquatic plant cover [Stewart 1988].

Often, when seafloor maps are generated, there are gaps in the data. This usually occurs because of time constraints on the sonar-mapping vessel or the navigation accuracy and maneuverability of the vessel when the sonar is in operation. The algorithm presented in this thesis can be adapted to handle data gaps in both the supplied map and the scanned profile. Instead of finding the MAD for all points of the profile, only the points for which a difference can be calculated (i.e. both a profile and a corresponding map point exist) contribute to the MAD value. The number of points can be different for different profile test locations at the same time step. Adjustments should then also be made to the weighting technique and the probabilities of detection and false alarm. Mathematically, this seems equivalent to changing the length of the profile. However, physically, biases can be introduced that unfairly favor matches with more or fewer points. If addressed properly, the solution to this problem would allow the AUV to navigate along the perimeter of the

map without biasing the updates toward the center of the map where the information is more complete.

Along different lines, future research could center on joining the terrain-relative navigation system with a mapping process [Leonard; Singh; Stewart 1988; Tuohy]. If the mapping routine were continually active, navigation could be achieved in a dynamic environment. The mapping system would track the temporal bathymetric changes and update the stored map as necessary. One step further would be to increase the accuracy or resolution of the stored map as the vehicle travels within the area. This is concurrent navigation and mapping.

The dilemma here is that to navigate well, a good map must be supplied, but the construction of a good map is dependent on accurate navigation. Most research to date has concentrated on the navigation and map-making capabilities separately, and two uncoupled processes are employed when a mission requires both. To my knowledge, the issues of concurrent localization and mapping in an underwater environment, including the bootstrapping problem, have not yet been addressed.

References

[1] Aguirre, F., J.M. Boucher, and J.J. Jacq, "Underwater Navigation by Video Sequence Analysis," *Proc. of the 10th ICPR,* Atlantic City, NJ, pp. 537-39, 1990.

[2] Ayache, N. and O. Faugeras, "Maintaining Representations of the Environment of a Mobile Robot," *IEEE Transactions on Robotics and Automation,* vol. 5, no. 6, pp 804-819, 1989.

[3] Bar-Shalom, Yaakov, ed., *Multitarget Multisensor Tracking: Advanced Applications,* Artech House, Norwood, MA, 1990.

[4] Bar-Shalom, Yaakov and Thomas E. Fortmann, *Tracking and Data Association.* Academic Press Inc., Boston, MA, 1988.

[5] Bellingham, J.G., "Capabilities of Autonomous Underwater Vehicles," in J. Moore, ed., *Scientific and Environmental Data Collection with Autonomous Underwater Vehicles,* pp. 7-14, Cambridge, MA, March, 1992a.

[6] Bellingham, J.G., "Review of Autonomous Underwater Vehicles", Massachusetts Institute of Technology Sea Grant Internal Memo IM34D1, 1992b.

[7] Bellingham, J.G., T.R. Consi, U. Tedrow, and D. Di Massa, "Hyperbolic Acoustic Navigation for Underwater Vehicles: Implementation and Demonstration," *Proc. AUV '92,* pp. 304-09, 1992.

[8] Bergem, Oddbjorn, *Bathymetric Navigation of Autonomous Underwater Vehicles Using a Multibeam Sonar and a Kalman Filter with Relative Measurement Covariance Matrices,* Sc. D. Thesis, Department of Informatics and Computer Science, University of Trondheim, Norway, 1993.

[9] Bernsten, M., "Principles and Practice of Acoustic Navigation Systems," in *International Underwater System Design,* vol. 14, No. 1, 1992.

[10] Blackington, J.G., D.M. Hussong, and J.G. Kosalas, "First Results from a Combination Sidescan Sonar and a Seafloor Mapping System (SeaMARC II)," in *Proc. of the 15th Annual Offshore Technology Conference,* vol. 1, pp. 307-14, Houston, TX, May, 1983.

[11] Bradley, A.M., "Low Power Navigation and Control for Long Range Autonomous Underwater Vehicles."

[12] Brown, Robert G. and Patrick Y.C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering, 2nd ed.,* John Wiley and Sons, Inc., New York, 1983.

[13] Busby Associates, Inc., *Undersea Vehicles Directory,* Arlington, VA, 1985.

[14] Calcar, Henry Van, "What You Need to Know to Improve LBL Submersible Tracking and Performance," *Proc. ROV '85*, pp. 43-51, 1985.

[15] Chantler, M.J., D.B. Lindsay, C.S. Reid, and V.J.C. Wright, "Optical and Acoustic Range Sensing for Underwater Robotics", *OCEANS '94*, pp. 205-210, 1994.

[16] Chen, D.J. and E. Alvarado, "Ultrashort Baseline Navigation for the Sea Squirt AUV," *Proc. of the 7th Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 644-49, Durham, NH, 1991.

[17] Cohen, P.M., *Bathymetric Navigation and Charting*, US Naval Institute, Annapolis, MD, 1970.

[18] Connell, J., "Controlling a Mobile Robot Using Partial Representations," SPIE '91: Mobile Robots, pp. 34-45, 1991.

[19] Cox, I.J. and G.T. Wilfong, eds., *Autonomous Robot Vehicles*, New York, Springer-Verlag, 1990.

[20] Deffenbaugh, Max, *A Matched Field Processing Approach to Long Range Acoustic Navigation*. MS Thesis, Massachusetts Institute of Technology/Woods Hole Oceanographic Institution Joint Program, 1994.

[21] Deffenbaugh, Max, *Optimal Ocean Acoustic Tomography with Moving Sources,"*, Ph.D. Thesis, Massachusetts Institute of Technology/Woods Hole Oceanographic Institution Joint Program, 1997.

[22] de Moustier, C. and H. Matsumoto, "Seafloor Acoustic Remote Sensing with Multibeam Echo-Sounders and Bathymetric Sidescan Sonar System, *Marine Geophysical Research*, vol. 15, no. 1, pp. 27-42, 1993.

[23] Dierckx, P., "An Algorithm for Surface-Fitting with Spline Functions," *IMA Journal of Numerical Analysis*, vol. 1, pp. 267-83, 1981.

[24] Di Massa, Diane E, *Operation of an Autonomous Underwater Vehicle Using Hyperbolic Navigation*, Mechanical Engineer Thesis, Dept. of Mechanical Engineering, Massachusetts Institute of Technology, 1992.

[25] Drumheller, M., "Mobile Robot Localization Using Sonar," Massachusetts Institute of Technology, AI Lab. Memo 826, Jan. 1985.

[26] Dyer, I. Personal communication, March 1994.

[27] Elfes, A., "Sonar-Based Real-World Mapping and Navigation," *Autonomous Robot Vehicles*, New York, Springer-Verlag, pp. 233-49, 1990a.

[28] Elfes, A., "Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception," *Proc. of 6th Conference on Uncertainty in Artificial Intelligence,* July, 1990b.

[29] Engleson, S. and D. McDermott, "Error Correction in Mobile Robot Map Learning," *Proc. of 1992 IEEE Intl. Conference on Robotics and Automation,"* Nice, France, pp. 2555-60, May 1992.

[30] Fortmann, T.E., Y. Bar-Shalom, and M. Scheffe, "Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association", IEEE Journal of Oceanic Engineering, vol. OE-8, pp. 173-84, July 1983.

[31] Foxwell, D. and M. Hewish. "Naval Navigation, the Right Course to Steer," in *International Defense Review,* No. 11, vol. 25, 1992.

[32] Gat, E., "On the Role of Theory in the Control of Autonomous Mobile Robots," Working Notes: *AAAI Fall Symposium Series on Applications of Artificial Intelligence to Real World Autonomous Mobile Robots,* Cambridge, MA, pp. 45-49, Oct. 1992.

[33] Gelb, Arthur, ed., *Applied Optimal Estimation,* The MIT Press, Cambridge, MA, 1974.

[34] Geyer, E.M., P. M. Creamer, J.A. D'Appolito, and R.G. Gains, "Characteristics and Capabilities of Navigation Systems for Unmanned Untethered Submersibles," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology,* pp. 320-47, Durham, NH, June, 1987.

[35] Golden, J.P., "Terrain Contour Matching (TERCOM): A Cruise Missile Guidance Aid," *SPIE: Image Processing for Missile Guidance,* vol. 238, pp. 10-18, 1980.

[36] Guibas, L. and L. Ramshaw, *Computer Graphics: Mathematical Foundations,* Course Notes CS348a, Stanford University, Department of Computer Science, 1992.

[37] Hebert, Martial, "Terrain Modeling for Autonomous Underwater Navigation," *Proc. 6th Intl. Symp. on Unmanned Untethered Submersible Technology,* pp. 502-11, Durham, NH, June, 1989.

[38] Hunter, G. and K. Steiglitz, "Operations on Images Using Quad-Trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. PAMI-1, no. 2, pp. 145-53, 1979.

[39] Jircitano, A., J. White, and D. Dosch, "Gravity Based Navigation of AUVs," *Proc. of the Symp. on Autonomous Underwater Vehicle Technology,* pp. 177-80, Washington DC, June 1990.

[40] Jolion, J.M. and A. Montanvert, "The Adaptive Pyramid: A Framework for 2D Image Analysis," *CVGIP: Image Understanding,* vol. 55, no. 3, pp. 339-48, May, 1992.

[41] Kamgar-Parsi, B., L.J. Rosenblum, F.J. Pipitone, L.S. Davis, and J.L. Jones, "Toward an Automated System for a Correctly Registered Bathymetric Chart," *IEEE Journal of Oceanic Engineering,* vol. 14, no. 4, pp. 314-25, October 1989.

[42] Kuipers, B.J. and Y.T. Byun, "A Robot Exploration and Mapping Strategy Based on a Semantic Hierarchy of Spatial Representations," AI90-120, Dept. Of Computer Science, Univ. of Texas at Austin, Jan. 16, 1990.

[43] Kuipers, B., R. Froom, W.-Y. Lee, and D. Pierce, "The Semantic Hierarchy Approach to Robot Learning," Working Notes: *AAAI Fall Symposium Series on Applications of Artificial Intelligence to Real World Autonomous Mobile Robots,* Cambridge, MA, pp. 90-97, October 1992.

[44] Kullander, L., "Development of a Terrain Navigation System for AUVs," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology,* pp. 494-501, Durham, NH, June 1989.

[45] Kwak, S.H., C.D. Stevens, J.R. Clynch, R.B. McGhee, and R.H. Whalen, "An Experimental Investigation of GSP/INS Integration for Small AUV Navigation," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology,* pp. 239-51, Durham, NH, Sept. 1993.

[46] Langer, D. and M. Hebert, "Building Elevation Maps from Underwater Sonar Data," *Proc. 7th Intl. Symp. on Unmanned Untethered Submersible Technology,* Merrimack, NH, pp. 328-41, 1991.

[47] Leader, Daniel Eugene, *Kalman Filter Estimation of Underwater Vehicle Position and Attitude Using a Doppler Velocity Aided Inertial Motion Unit,* Ocean Engineer Thesis, Massachusetts Institute of Technology/Woods Hole Oceanographic Institution Joint Program, 1994.

[48] Leonard, J.J., *Directed Sonar Sensing for Mobile Robot Navigation,* Ph.D. Thesis, University of Oxford, 1990.

[49] Leonard, J.J. and H.F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation.* Robotics: Vision, Manipulation, and Sensors, Kluwer Academic, Boston, MA, 1992.

[50] Lerro, Donald and Yaakov Bar-Shalom, "Automated Tracking with Target Amplitude Information," *Proc. of the 1990 American Control Conference,* pp. 2875-80, San Diego, CA, May 23-25, 1990.

[51] Lim, Jae S., *Two-Dimensional Signal and Image Processing*, Prentice Hall, Englewood Cliffs, NJ, 1990.

[52] Marquest Inc., Marine Telepresence Division, Specification of the Sonic High Accuracy Ranging and Positioning System (SHARPS), Bourne, MA, 1989.

[53] Matthies, L., and A. Elfes, "Integration of Sonar and Stereo Range Data using a Grid-based Representation," *Proc. IEEE Intl. Conference on Robotics and Automation*, pp. 727-33, 1988.

[54] Moravec, H. and M. Blackwell, "Learning Sensor Models for Evidence Grids," Carnegie-Melon University, Robotics Institute Annual Research Review, pp. 8-15, 1993.

[55] Nodland, W.E., "A General Description of the Self-Propelled Underwater Research Vehicle (SPURV)," APL-UW 6814, Applied Physics Laboratory, University of Washington, 1968.

[56] Orser, D.J. and M. Roche, "The Extraction of Topographic Features in Support of Autonomous Underwater Vehicle Navigation," *Proc. 5th Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 502-14, Merrimack, NH, June, 1987.

[57] Parker, R.L. and S.P. Heustis, "The Inversion of Magnetic Anomalies in the Presence of Topography," *Journal. of Geophysical Research*, vol. 79, pp. 1587-93, 1974.

[58] Papoulis, Athanasios, *Probability, Random Variables and Stochastic Processes, 3rd ed.*, McGraw-Hill Inc., New York, 1991.

[59] Pierce, J.A., "An Introduction to LORAN," *IEEE AES Magazine*, pp. 16-33, Oct., 1990.

[60] Pierce, J.A., A.A. McKenzie, and R.H. Woodward, eds., *LORAN, Long Range Navigation*, McGraw-Hill Book Co., New York, NY, 1948.

[61] Polvani, D., "Magnetic Guidance of Autonomous Vehicles, Part II," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 257-64, Durham, NH, Sept. 1987.

[62] RD Instruments Inc.,. "Technical Specifications of the RDI Workhorse Navigator Doppler Velocimeter Log," San Diego, CA, 1996.

[63] Rao, B.M. L., W.H. Hoge, and R.P. Hamlen, "An Advanced Aluminum Power Source for UUVs," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 71-78, Durham, NH, Sept. 1991.

[64] Romesburg, H.C., *Cluster Analysis for Researchers,* Lifetime Learning Publications, Belmont, CA, 1984.

[65] Singh, H., *An Entropic Framework for AUV Sensor Modelling,* Ph.D. Thesis, Massachusetts Institute of Technology/Woods Hole Oceanographic Institution Joint Program, 1995.

[66] Singh, H., D. Yoerger, R. Bachmayer, A. Bradley, and W.K. Stewart, "Sonar Mapping with the Autonomous Benthic Explorer (ABE)," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology,* pp. 367-75, Durham, NH, Sept. 1995.

[67] Smith, R., M. Self, and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics," in I. Cox and G. Wilfong, eds., *Autonomous Robot Vehicles,* Springer-Verlag, New York, 1990.

[68] Stewart, W.K. Personal communications, February 1997.

[69] Stewart, W.K., "High Resolution Optical and Acoustic Remote Sensing for Underwater Exploration," *Oceanus,* vol. 34, no. 1, pp. 10-22, Spring 1991.

[70] Stewart, W.K., *Multisensor Modelling Underwater with Uncertain Information,* Ph.D. Thesis, Massachusetts Institute of Technology/Woods Hole Oceanographic Institution Joint Program, 1988.

[71] Thomas, George B., Jr. and Ross L. Finney, *Calculus and Analytic Geometry, 5th ed.,* Addison-Wesley Publishing Co., Reading, MA, 1982.

[72] Tracey, Brian Hearne, *Design and Testing of an Acoustic Ultra-short Baseline Navigation System,* Massachusetts Institute of Technology/Woods Hole Oceanographic Institution Joint Program, 1992.

[73] Tsipis, K., "Cruise Missiles," *Scientific American,* vol. 236, no. 2, pp. 20-29, February 1977.

[74] Tuohy, S.T., *Geophysical Map Representation, Abstraction, and Interrogation for Autonomous Underwater Vehicle Navigation,* Ph.D. Thesis, Department of Ocean Engineering, Massachusetts Institute of Technology, 1993.

[75] Turner, R.M., "Context-Sensitive, Adaptive Reasoning for Intelligent AUV Control; ORCA Project Update," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology,* pp. 426-35, Durham, NH, Sept. 1995.

[76] Tyren, C. "Magnetic Anomalies as a Reference for Ground-Speed and Map-Matching Navigation," *Journal. of Navigation,* vol. 35, No. 2, pp. 242-54, May 1982.

[77] Tyren, C., "Magnetic Terrain Navigation," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 245-56, Durham, NH, Sept. 1987.

[78] Uhrich, R. and J. Watson, "Supervisory Control of the Untethered Undersea Systems: A New Paradigm Verified," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 1-5, Durham, NH, Sept. 1995.

[79] Van Trees, H.L., *Detection, Estimation, and Modulation Theory*, John Wiley and Sons, New York, 1972.

[80] Walter, C.E., "Infrastructure for Thulium-170 Isotope Power Systems for Autonomous Underwater Vehicle Fleets," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 79-88, Durham, NH, Sept. 1991.

[81] Waterman, T.H., *Animal Navigation*, Scientific American Library, 1989.

[82] Watkinson, K.W., D.E. Humphreys, and N.S. Smith, "UUV Controllability at Transition Speeds: Thruster and Fin Control Authority Requirements," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 97, Durham, NH, Sept. 1995.

[83] Willen, S., "Cryogenic Oxygen System for AUV Power Applications," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 89-103, Durham, NH, Sept. 1991.

[84] Willsky, Alan S., Gregory W. Wronell, and Jeffrey H. Shapiro, *Stochastic Processes, Detection and Estimation: 6.432 Supplementary Course Notes*, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA, 1995.

[85] Winston, Patrick, *Artificial Intelligence, 3rd ed.*, Addison-Wesley Publishing Co., Reading, MA, 1992.

[86] Xu, M., and S.E. Smith, "A Fuzzy Sliding Controller for Autonomous Underwater Vehicles," in *Proc. Intl. Symp. on Unmanned Untethered Submersible Technology*, pp. 178-85, Durham, NH, Sept. 1995.
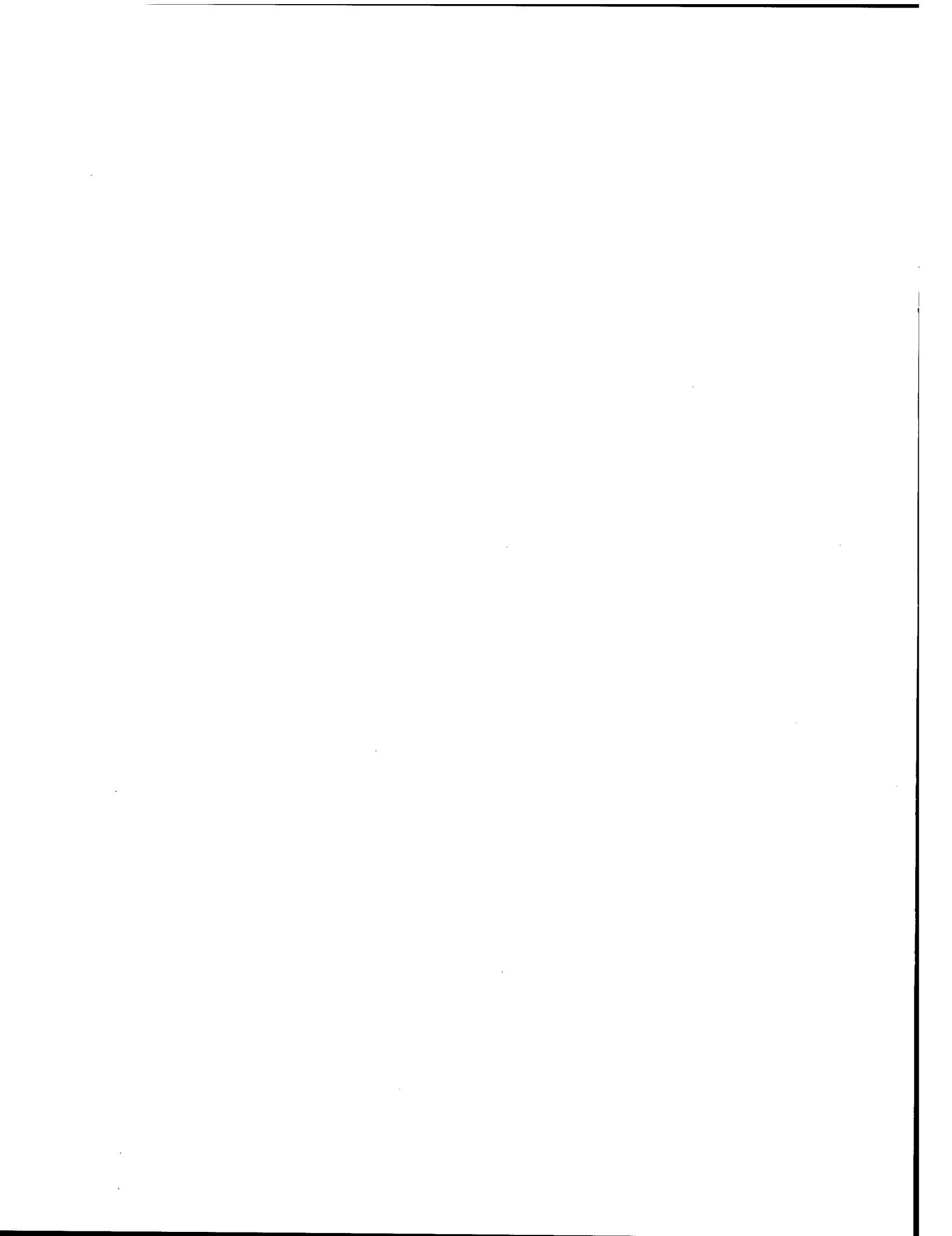
DOCUMENT LIBRARY
Distribution List for Technical Report Exchange – February 1996

University of California, San Diego
SIO Library 0175C
9500 Gilman Drive
La Jolla, CA 92093-0175

Hancock Library of Biology & Oceanography
Alan Hancock Laboratory
University of Southern California
University Park
Los Angeles, CA 90089-0371

Gifts & Exchanges
Library
Bedford Institute of Oceanography
P.O. Box 1006
Dartmouth, NS, B2Y 4A2, CANADA

Commander
International Ice Patrol
1082 Shennecossett Road
Groton, CT 06340-6095

NOAA/EDIS Miami Library Center
4301 Rickenbacker Causeway
Miami, FL 33149

Research Library
U.S. Army Corps of Engineers
Waterways Experiment Station
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

Institute of Geophysics
University of Hawaii
Library Room 252
2525 Correa Road
Honolulu, HI 96822

Marine Resources Information Center
Building E38-320
MIT
Cambridge, MA 02139

Library
Lamont-Doherty Geological Observatory
Columbia University
Palisades, NY z10964

Library
Serials Department
Oregon State University
Corvallis, OR 97331

Pell Marine Science Library
University of Rhode Island
Narragansett Bay Campus
Narragansett, RI 02882

Working Collection
Texas A&M University
Dept. of Oceanography
College Station, TX 77843

Fisheries-Oceanography Library
151 Oceanography Teaching Bldg.
University of Washington
Seattle, WA 98195

Library
R.S.M.A.S.
University of Miami
4600 Rickenbacker Causeway
Miami, FL 33149

Maury Oceanographic Library
Naval Oceanographic Office
Building 1003 South
1002 Balch Blvd.
Stennis Space Center, MS, 39522-5001

Library
Institute of Ocean Sciences
P.O. Box 6000
Sidney, B.C. V8L 4B2
CANADA

National Oceanographic Library
Southampton Oceanography Centre
European Way
Southampton SO14 3ZH
UK

The Librarian
CSIRO Marine Laboratories
G.P.O. Box 1538
Hobart, Tasmania
AUSTRALIA 7001

Library
Proudman Oceanographic Laboratory
Bidston Observatory
Birkenhead
Merseyside L43 7 RA
UNITED KINGDOM

IFREMER
Centre de Brest
Service Documentation - Publications
BP 70 29280 PLOUZANE
FRANCE

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO. MIT/WHOI 97-22	2.	3. Recipient's Accession No.
4. Title and Subtitle Terrain-Relative Navigation for Autonomous Underwater Vehicles			5. Report Date June 1997
			6.
7. Author(s) Diane E. Di Massa			8. Performing Organization Rept. No.
9. Performing Organization Name and Address MIT/WHOI Joint Program in Oceanography/Applied Ocean Science & Engineering			10. Project/Task/Work Unit No. MIT/WHOI 97-22
			11. Contract(C) or Grant(G) No. (C) N00014-96-1-5028; N00014-94-1-0466; N00017-92-J-1714; N00140-90-D-1979; (G) N00014-92-C-6028
12. Sponsoring Organization Name and Address			13. Type of Report & Period Covered Ph.D. Thesis
			14.

15. Supplementary Notes

This thesis should be cited as: Diane E. Di Massa, 1997. Terrain-Relative Navigation for Autonomous Underwater Vehicles. Ph.D. Thesis. MIT/WHOI, 97-22.

16. Abstract (Limit: 200 words)

This thesis presents a terrain-relative navigation system for AUVs that does not require the deployment of acoustic beacons or other navigational aids, but instead depends on a supplied digital bathymetric map and the ability of the vehicle to image the seafloor.

At each time step, a bathymetric profile is measured and compared to a local region of the supplied map using a mean absolute difference criterion. The region size is determined by the current navigation uncertainty. For large regions, a coarse-to-fine algorithm with a modified beam search is used to intelligently search for good matches while reducing the computational requirements. A validation gate is defined around the position estimate using the navigation uncertainty, which is explicitly represented through a covariance matrix. A probabilistic data association filter with amplitude information (PDAFAI), grounded in the Kalman Filter framework, probabilistically weights each good match that lies within the validation gate. Weights are a function of both the match quality and the size of the innovation. Navigation updates are then a function of the predicted position, the gate size, all matches within the gate, and the uncertainties on both the prediction and the matches.

The system was tested in simulation on several terrain types using a deep-ocean bathymetry map of the western flank of the Mid-Atlantic Ridge between the Kane and Atlantis Transforms. Results show more accurate navigation in the areas with greater bathymetric variability and less accurate navigation in flatter areas with more gentle terrain contours. In most places, the uncertainties assigned to the navigation positions reflect the ability of the system to follow the true track. In no case did the navigation diverge from the true track beyond the point of recovery.

17. Document Analysis   a. Descriptors

Navigation
Underwater
Vehicles

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement Approved for publication; distribution unlimited.	19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 154
	20. Security Class (This Page)	22. Price

(See ANSI-Z39.18)   *See Instructions on Reverse*   OPTIONAL FORM 272 (4-77)
(Formerly NTIS-35)
Department of Commerce